SynRS3D: A Synthetic Dataset for Global 3D Semantic Understanding from Monocular Remote Sensing Imagery

Jian Song[1,2], **Hongruixuan Chen**[1], **Weihao Xuan**[1,2], **Junshi Xia**[2], **Naoto Yokoya**[1,2]

[1]The University of Tokyo, Tokyo, Japan
[2]RIKEN AIP, Tokyo, Japan
song@ms.k.u-tokyo.ac.jp
https://JTRNEO.github.io/SynRS3D

Abstract

Global semantic 3D understanding from single-view high-resolution remote sensing (RS) imagery is crucial for Earth observation (EO). However, this task faces significant challenges due to the high costs of annotations and data collection, as well as geographically restricted data availability. To address these challenges, synthetic data offer a promising solution by being unrestricted and automatically annotatable, thus enabling the provision of large and diverse datasets. We develop a specialized synthetic data generation pipeline for EO and introduce *SynRS3D*, the largest synthetic RS dataset. SynRS3D comprises 69,667 high-resolution optical images that cover six different city styles worldwide and feature eight land cover types, precise height information, and building change masks. To further enhance its utility, we develop a novel multi-task unsupervised domain adaptation (UDA) method, *RS3DAda*, coupled with our synthetic dataset, which facilitates the RS-specific transition from synthetic to real scenarios for land cover mapping and height estimation tasks, ultimately enabling global monocular 3D semantic understanding based on synthetic data. Extensive experiments on various real-world datasets demonstrate the adaptability and effectiveness of our synthetic dataset and the proposed RS3DAda method. SynRS3D and related codes are available at https://github.com/JTRNEO/SynRS3D.

1 Introduction

3D reconstruction is a fundamental task in computer vision, which focuses on creating three-dimensional representations from two-dimensional images. It plays a crucial role in applications such as virtual reality, autonomous driving, and robotics. In the context of Earth observation (EO), reconstructing semantic 3D information from single-view remote sensing (RS) images is also vital for applications like environmental monitoring, urban planning, and disaster response [50, 52, 53, 65]. Unlike point cloud-based 3D reconstruction, which relies on LiDAR or stereo cameras, monocular semantic 3D reconstruction is more scalable and requires less expensive equipment, making it suitable for global applications [21, 87, 120, 44, 50, 52, 53, 65, 54, 26, 22, 49]. This task combines land cover mapping, which is closely related to semantic segmentation in computer vision, and height estimation. However, acquiring RS annotations is costly and time-consuming, especially for high-resolution height data obtained through satellite LiDAR (e.g. GEDI, ICESat-2) [85, 30, 51] or stereo matching [3, 116, 112, 57, 27, 64]. Furthermore, high-resolution land cover mapping datasets [16, 96, 101]

38th Conference on Neural Information Processing Systems (NeurIPS 2024) Track on Datasets and Benchmarks.

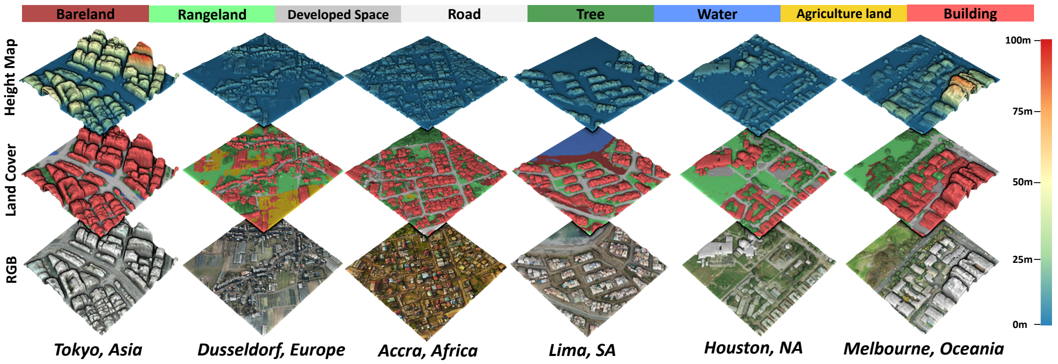


Figure 1: 3D visualization outcomes from real-world monocular RS images, using the model trained on the SynRS3D dataset with the proposed RS3DAda method. The top colorbar represents the legend for the land cover map (row 2), while the right colorbar indicates the height range (row 1). "SA" indicates South America and "NA" indicates North America.

often lack the corresponding height data. Moreover, the availability of RS data is geographically skewed, with developed regions having abundant data and developing regions lacking high-resolution datasets. Schmitt et al. [83] reviewed more than 380 RS datasets, revealing that few datasets come from Oceania, South America, Africa and Asia, while most originate from Europe and North America. This geographic limitation in RS datasets raises a crucial question: Can findings from numerous research papers be applied to these underrepresented regions?

The aforementioned challenges can be effectively addressed using synthetic data. Current 3D modeling technology has the potential to create various landscape features with accompanying land cover semantic labels and height values. Therefore, we present *SynRS3D*, a high-quality, high-resolution synthetic RS 3D dataset. SynRS3D includes 69,667 images with various ground sampling distances (GSD) ranging from 0.05 to 1 meter, and annotations for height estimation and land cover mapping in various scenes. However, models trained solely on synthetic data tend to overfit to these datasets, resulting in significantly reduced performance when applied to real-world environments due to the large domain gap. Existing synthetic datasets [7, 124, 43, 84, 105, 75, 76, 86, 104] often exhibit this significant performance gap compared to models trained on real data.

To bridge this gap, we introduce *RS3DAda*, a novel baseline aimed at advancing research on SynRS3D and setting a benchmark for multi-task unsupervised domain adaptation (UDA) from synthetic to real scenarios of RS. This approach utilizes a self-training framework and incorporates a land cover branch to enhance the quality of pseudo-labels of the height estimation branch, thus stabilizing RS3DAda training and boosting the accuracy of both branches. For the height estimation task, our model even outperforms models trained on real-world data in the challenging areas. Figure 1 shows the results of the 3D semantic reconstruction globally using models trained with our RS3DAda method on the SynRS3D dataset. Furthermore, we include disaster mapping results for earthquake and wildfire scenarios using our model in Appendix A.9, showcasing its efficacy in real-world disaster response applications.

The major contributions of this work can be summarized in three aspects:

- We propose **SynRS3D**, the largest RS synthetic dataset with comprehensive annotations and geographic diversity for remote sensing tasks.
- We design **RS3DAda**, a robust and effective multi-task UDA algorithm for land cover mapping and height estimation.
- Based on SynRS3D, we benchmark various remote sensing scenarios for land cover mapping and height estimation tasks.

We hope that our dataset and the proposed method advance the progress of synthetic learning in remote sensing applications.

2 Related Work

High-Resolution Earth Observation. High-resolution RS technology enables us to capture images with a GSD of less than 1 meter, significantly enhancing our understanding of Earth's surface details. Deep learning has become a powerful tool for analyzing these images. High-resolution imagery allows for precise land cover mapping [55, 121, 34, 97, 60, 102]. Concurrently, height estimation research [54, 26, 22, 49] focuses on determining accurate surface elevations. Some studies [21, 87, 120, 44] combine these tasks, training models for both land cover mapping and height estimation simultaneously. Most 3D reconstruction studies use real-world data, concentrating on buildings and often neglecting valuable classes like trees [50, 52, 53, 65]. Multi-view RS for 3D reconstruction [81, 20, 38, 19, 113, 47] is expensive and geographically limited. Benchmark datasets [16, 96, 101, 106, 45, 73, 13, 80] have been constructed for model training; however, acquiring high-resolution RS data is costly and time-consuming due to manual labeling and sophisticated equipment requirements. This limits the number of samples available to train robust models. Additionally, real datasets often lack geographic diversity, which can hinder model generalization to new, unseen areas.

Synthetic Remote Sensing Dataset. Modern methods to create synthetic data utilize deep learning generative models, such as diffusion models [31] and generative adversarial networks (GANs) [25], in conjunction with 3D modeling techniques. Generative models often struggle to produce data outside their training distribution and lack the precise control needed for RS tasks. In contrast, 3D modeling approaches in computer graphics, which take advantage of game engines or 3D software [8, 66, 79, 77, 28, 100], have shown more success. However, creating synthetic data for RS is inherently more complex. A single 1024x1024 RS image demands hundreds of buildings, thousands of trees, and various topological features such as rivers and roads, unlike street views that require fewer assets. Recent studies have used 3D software to synthesize RS data with automatic annotations [7, 124, 43, 84, 105, 75, 76, 86, 104]. Despite their utility, these datasets often suffer from limited geographic diversity [7, 124, 105, 75, 76, 104], semantic categories [7, 124, 43, 84, 105, 75, 76, 104], and comprehensive semantic and height information [7, 124, 43, 84, 105, 86], which impacts their effectiveness in training robust, globally applicable models.

Unsupervised Domain Adaptation (UDA). UDA aims to adapt a model trained on labeled data from a source domain to perform well in an unlabeled target domain, reducing the constraints and costs of data annotation. For semantic segmentation, most of the work focuses on adversarial learning [92, 93, 33, 32, 23, 98, 95, 91, 63] and self-training [122, 56, 115, 123, 67, 82, 110, 94, 35, 36, 37, 48, 24]. Unlike semantic segmentation tasks, height estimation is a regression task, aligning closely with monocular depth estimation. For UDA in monocular depth estimation, methods often focus on image translation to reduce domain gaps [4, 117, 119, 12], and some use self-training with pseudo-labels [61, 107, 111]. In RS, most research [40, 62, 72, 89, 90, 118] focuses on applying developed techniques from computer vision to adapt models trained on real-world data to different real-world environments (real-to-real). However, only a small number of studies [58, 103] investigate the challenges of adapting synthetic data to real-world environments (synthetic-to-real). To the best of our knowledge, RS3DAda is the first work to explore UDA algorithms specifically designed for synthetic-to-real domain adaptation for multi-task dense prediction in RS.

3 The SynRS3D Acquisition Protocol and Statistical Analysis

Although simulating RS scenes poses significant challenges due to the need for numerous assets, we mitigate this issue using procedural modeling techniques [69, 68, 42]. Instead of manually modeling each element, we incorporate rules derived from real-world knowledge, formalized into scripts. The generation system is controlled through hyperparameters like city style, asset ratio, and texture style, allowing us to create a diverse high-quality synthetic dataset. Table 1 compares existing synthetic RS datasets with SynRS3D, highlighting our dataset's advantages in diversity, functionality, and scale.

3.1 Statistics for SynRS3D

The left section of Figure 2 shows RGB images, land cover labels, and height maps for six styles in SynRS3D, representing diverse real-world environments. The bottom left section compares the height distribution of SynRS3D with two leading synthetic datasets, SMARS [76] and GTAH [104],

Table 1: Comparisons of various RS synthetic datasets based on diversity, image capture details, asset origins, tasks, and the number of images. The diversity is categorized into City-Replica (datasets mimicking specific cities) and Style-Extended (covering a range of urban styles). Image capture attributes include GSD, resolution, and perspective (Nadir, Oblique). Asset origins are denoted as Manually-made (M), Game-origin (G), Procedurally-generated (P), and Real (R). Tasks cover Change Detection (CD), Building Segmentation (BS), Object Detection (OD), Disparity Estimation (DE), Height Estimation (HE), Land Cover (LC), and Building Change Detection (BCD).

RS Synthetic Datasets	Diversity		Image Capture			Assets			Task	# Images
	City-Replica	Style-Extended	GSD (m)	Image Size	Perspective	Layout	Geometry	Texture		
AICD [7]	✓	✗	$*$	800×600	Nad., Obl.	M	M	M	CD	$\sim 1K$
GTA-V-SID [124]	✓	✗	1.0	500×500	Nad.	G	G	G	BS	$\sim 0.12K$
Synthinel-1 [43]	✓	✓	0.3	572×572	Nad.	R	M	M+R	BS	$\sim 1K$
RarePlanes [84]	✓	✗	$0.31 - 0.39$	512×512	Nad., Obl.	R	M	M+R	OD	$\sim 65K$
SyntCities [75]	✓	✗	$0.1, 0.3, 1.0$	1024×1024	Nad.	R	M	M+R	DE	$\sim 8K$
GTAH [104]	✓	✗	$*$	1920×1080	Nad., Obl.	G	G	G	HE	$\sim 28.6K$
SyntheWorld [86]	✗	✓	$0.3 - 1.0$	Various	Nad., Obl.	P	P+M	P	LC, BCD	$\sim 40K$
SMARS [76]	✓	✗	$0.3, 0.5$	Various	Nad.	R	M	M+R	LC, HE, BCD	4
SynRS3D (Ours)	✗	✓	$0.05 - 1.0$	512×512	Nad., Obl.	P	P+M	P	LC, HE, BCD	$\sim 70K$

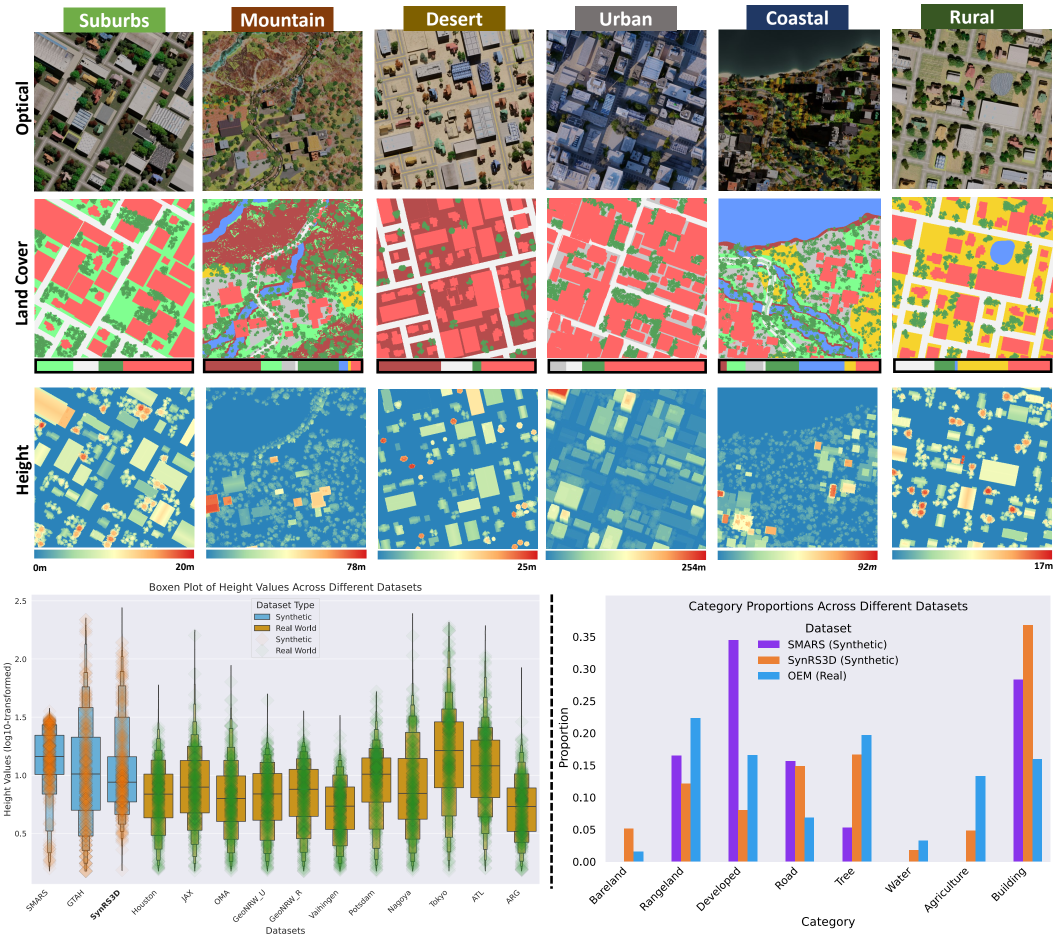


Figure 2: Examples and statistics of SynRS3D. The colorbar corresponds to the land cover classification legend shown in Figure 1.

as well as 11 real-world height datasets. SynRS3D's height distribution closely matches real-world data, while SMARS and GTAH show limitations. Specifically, SMARS, which mimics Paris and Venice, has a narrow height range. GTAH, based on the GTAV game, which mimics Los Angeles, shows a wider height range, but with a larger mean and variance, making it less representative of other cities. SynRS3D was constructed using the following prior knowledge [99]: backward regions (low buildings) cover about 12% of the world's areas, emerging regions (mid buildings) cover about

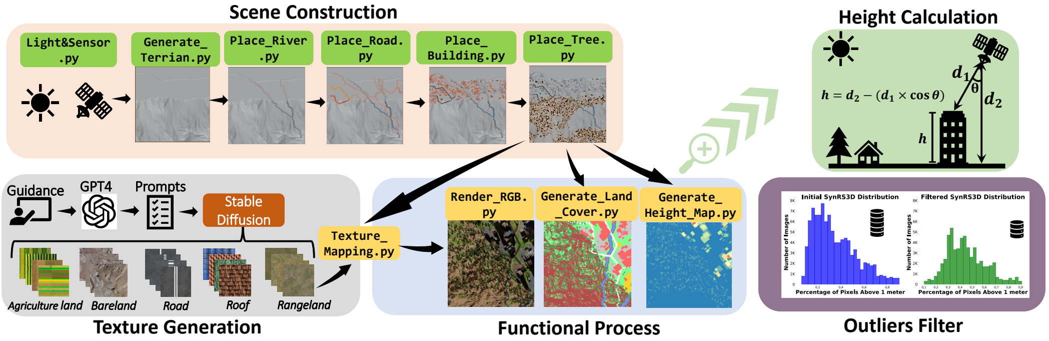


Figure 3: Generation workflow of SynRS3D.

70%, and developed regions (tall buildings) cover about 18%. The bottom right section contrasts the land cover proportions in SynRS3D with those in SMARS and the real-world OEM dataset [101], which is currently the largest and most geographically diverse dataset. SynRS3D's land cover categories—including Bareland, Rangeland, Developed Space, Roads, Trees, Water, Agricultural Land, and Buildings—match well with the OEM dataset. In contrast, SMARS has only five categories, limiting its effectiveness for comprehensive land cover mapping.

3.2 Generation Workflow of SynRS3D

The generation process of SynRS3D employs tools such as Blender [14], Python, GPT-4 [2], and Stable Diffusion [78], as illustrated in Figure 3. It begins with Python scripts that translate synthetic scene rules into parameter-controlled instructions for tasks like terrain generation, sensor placement, and asset placement. The geometry of the buildings and trees is created both procedurally and manually. Stable Diffusion generates textures based on detailed text prompts from GPT-4, ensuring high-quality and diverse textures. Blender's compositor node and Python scripts then generate accurate land cover labels and height maps. The details of the height map generation are detailed in the height calculation section of Figure 3. Our dataset's height maps are produced within Blender using simple geometric algorithms, resulting in a completely accurate normalized Digital Surface Model (nDSM). An nDSM represents the height of objects above the ground, providing clear information about buildings and vegetation. In contrast, the height maps for the comparative datasets GTAH and SMARS are Digital Surface Models (DSM), which include the height of ground and objects. Converting DSM to nDSM requires additional processing using the dsm2dtm [1] algorithm, which introduces noises. Optical images are produced using rendering scripts. To generate building change detection masks, we follow a structured process. Initially, buildings are randomly removed from scenes, and textures are reapplied to create pre-event images. Subsequently, land cover labels are subtracted to produce the change detection masks. After the initial generation of the dataset, images with anomalous height distributions are filtered out. This step ensures that the final version of SynRS3D closely aligns with real-world height distributions. The specific filtering algorithm used and examples of building change detection masks can be found in the Appendix A.1.

4 Multi-Task Unsupervised Domain Adaptation for RS Tasks (RS3DAda)

SynRS3D features low costs, high diversity, and large volume. However, there is a clear domain gap between synthetic data and real-world environments, which limits the use of SynRS3D. This limitation is particularly evident in RS, where synthetic-to-real UDA algorithms are lacking. To bridge this gap, we developed RS3DAda, which leverages land cover labels and height values to complement each other. In addition, it harnesses the potential of unlabeled real-world data, establishing the first benchmark for synthetic-to-real RS-specific multi-task UDA.

[1] https://github.com/seedlit/dsm2dtm

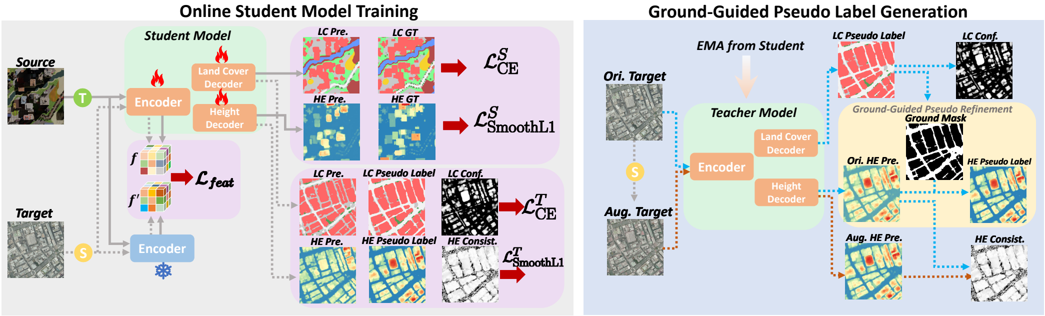


Figure 4: Overview of the proposed RS3DAda method. T denotes statistical image translation, S represents strong augmentation. For Online Student Model Training, dotted line: target image, solid line: source image. For Ground-Guided Pesudo Label Generation, dotted line: original target image, dotted line: strong augmented target image.

Basic Framework. In this work, we adopt self-training [46] as our basic UDA technique due to its superior stability and adaptability across both land cover mapping and height estimation tasks. Additionally, the teacher-student framework [88] is incorporated into the method to enhance performance and generalizability by stabilizing training with the exponential moving average (EMA) of model weights. The synthetic dataset image set $\mathcal{X}_s$ serves as the source domain, with access to corresponding land cover labels $\mathcal{Y}^s_{LC}$ and height maps $\mathcal{Y}^s_{H}$. The real dataset image set $\mathcal{X}_t$ serves as the target domain without any access to the labels.

Source Domain Training. In this stage, shown in the left part of Figure 4, source domain images $\mathcal{X}_s$ undergo statistical image translation using simple Fourier Domain Adaptation (FDA) [110], Histogram Matching (HM) and Pixel Distribution Matching (PDA), which follow the conclusion of Abramov et al. [1]. This process aligns the styles of the source images with the target domain, resulting in translated images $\mathcal{X}'_s$. These translated images from the source domain are then fed into the student network, producing predicted land cover labels $\hat{Y}^s_{LC}$ and height $\hat{Y}^s_{H}$. The supervised loss for the source domain is defined as:

$$\mathcal{L}_{source} = \frac{1}{N}\sum_{i=1}^{N}\left(\mathcal{L}_{CE}(\hat{Y}^{s(i)}_{LC}, \mathcal{Y}^{s(i)}_{LC}) + \mathcal{L}_{SmoothL1}(\hat{Y}^{s(i)}_{H}, \mathcal{Y}^{s(i)}_{H})\right), \tag{1}$$

where $\mathcal{L}_{CE}$ is the cross-entropy loss, $\mathcal{L}_{SmoothL1}$ is the Smooth L1 loss [39], and N is the total number of samples.

Land Cover Pseudo-Label Generation. To generate high-quality pseudo-labels for land cover mapping, as illustrated in the right section of Figure 4, target domain images $\mathcal{X}_t$ are strongly augmented denoted as $\mathcal{X}'_t$. We adopt color jitter, Gaussian blur, and ClassMix [70] as the strong augmentations. The teacher model then predicts the land cover pseudo-labels $\tilde{Y}^t_{LC}$. After that, we use a threshold τ to generate a confidence map $\mathcal{C}_{LC} = \mathbb{I}(\tilde{Y}^t_{LC} > \tau)$, where $\mathbb{I}$ is the indicator function.

Height Pseudo-Label Generation. Height pseudo-labels are generated by leveraging prior knowledge that only trees and buildings have height values. This is the first attempt to use ground information to correct height pseudo-labels, inspired by the empirical observations that the network achieves superior accuracy on the ground class early in the training stage. We refine height pseudo-labels using a ground mask $\mathcal{G}$ generated from the land cover mapping branch as $\mathcal{G} = \mathbb{I}(\tilde{Y}^t_{LC} = \text{Ground})$, and refine the height pseudo-labels by $\tilde{Y}^{t,refined}_{H} = \tilde{Y}^{t,ori}_{H} \cdot (1 - \mathcal{G})$. We also create a height consistency map $\mathcal{C}_H$:

$$\mathcal{C}_H = \mathbb{I}\left(\max\left(\frac{\tilde{Y}^{t,ori}_{H}}{\tilde{Y}^{t,aug}_{H}}, \frac{\tilde{Y}^{t,aug}_{H}}{\tilde{Y}^{t,ori}_{H}}\right) \leq \eta\right), \tag{2}$$

indicating that we consider predictions reliable if they remain stable under perturbations, where η is the threshold.

Target Domain Training. The target domain training loss is:

$$\mathcal{L}_{target} = \frac{1}{N}\sum_{i=1}^{N}\left(\mathcal{L}_{CE}(\hat{Y}_{LC}^{t(i)}, \tilde{Y}_{LC}^{t(i)}) \cdot \mathcal{C}_{LC}^{(i)} + \mathcal{L}_{SmoothL1}(\hat{Y}_{H}^{t(i)}, \tilde{Y}_{H}^{t,refined(i)}) \cdot \mathcal{C}_{H}^{(i)}\right). \tag{3}$$

Feature Constraint. To alleviate overfitting, we adopt a frozen DINOv2 [71] encoder to supervise the student encoder's updates, inspired by Yang et al. [109]. Let $\mathbf{f}$ denote the features of the student encoder and $\mathbf{f}'$ denote the features of the frozen DINOv2 encoder. We utilize cosine similarity to constrain the feature updates with the loss defined as:

$$\mathcal{L}_{feat} = \begin{cases} 1 - \frac{\mathbf{f}\cdot\mathbf{f}'}{\|\mathbf{f}\|\|\mathbf{f}'\|} & \text{if } \frac{\mathbf{f}\cdot\mathbf{f}'}{\|\mathbf{f}\|\|\mathbf{f}'\|} < \epsilon \\ 0 & \text{otherwise} \end{cases}, \tag{4}$$

where ϵ is the threshold.

Overall Loss. The overall loss of the model is defined as the sum of the source domian training loss, target domain training loss, and feature alignment loss, each weighted by their respective coefficients. Formally, the overall loss is given by:

$$\mathcal{L}_{\text{overall}} = \mathcal{L}_{\text{source}} + \lambda_{\text{target}}\mathcal{L}_{\text{target}} + \lambda_{\text{feat}}\mathcal{L}_{\text{feat}}, \tag{5}$$

where λ_{target} and λ_{feat} are weighting coefficients that control the contribution of the target and feature alignment loss terms, respectively.

5 Experiments

Evaluation Datasets & Experimental Setting. We evaluate our synthetic dataset, SynRS3D, from various aspects. Table 2 (a) shows the real-world height estimation datasets, while Table 2 (b) lists the real-world land cover mapping datasets. In Section 5.1, we compare SynRS3D with other synthetic datasets under the source-only setting, a term commonly used in UDA to describe models trained solely on the source domain and directly applied to the target domain without adaptation. This setup highlights SynRS3D's smaller domain gap and its potential for direct usage in real-world scenarios. In Section 5.2, we investigate the advantages of SynRS3D for augmenting real-world data through fine-tuning and joint training. Specifically, in Section 5.3, to evaluate the effectiveness of RS3DAda, we divide the 11 height estimation datasets into two target domains: *Target Domain 1* includes 6 widely-used public datasets, while *Target Domain 2* contains 5 more challenging datasets to assess and contrast the generalization ability of SynRS3D with real datasets.

Evaluation Metrics. We employ Intersection over Union (IoU) and Mean Intersection over Union (mIoU) as the metrics to evaluate the model's performance for land cover mapping tasks. For height estimation, we use Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and accuracy metrics [18] denoted as δ, along with our custom metric F_1^{HE}. Detailed metric definitions are in Appendix A.3.

Implementation Details. Unless specifically detailed, all experiments utilize the pre-trained DINOv2 [71] implemented with ViT-L [17] as the encoder, with DPT [74] serving as both the land cover mapping and height estimation decoders. The hyperparameters for the various experiments are provided in Appendix A.4.

Table 2: Datasets setup for the experiments on height estimation and land cover mapping. (a) **Height Estimation Datasets**: We detail 11 height estimation datasets categorized into two target domains. The first six datasets for training the model with real-world data are sourced from Europe and the United States as shown in Section 5.3. The remaining five datasets covering more challenging areas are characterized by: notable height mean&standard deviation, non-RGB channels, and varied regions outside of the US and EU, used for evaluation in Section 5.3. (b) **Land Cover Mapping Datasets**: We evaluate our method on five commonly used datasets covering diverse environments.

Real-World Height Estimation Datasets

Types	Datasets	Region	Height mean&std	Channel
Target Domain 1	Houston [106]	US	[3.07, 5.02]	RGB
	JAX [45]	US	[4.73, 9.02]	RGB
	OMA [45]	US	[2.37, 5.27]	RGB
	GeoNRW_Urban [5]	Germany	[2.46, 4.31]	RGB
	GeoNRW_Rural [5]	Germany	[2.03, 4.21]	RGB
	Potsdam [80]	Germany	[3.02, 5.68]	RGB
Target Domain 2	ATL [13]	US	[8.40, 13.41]	RGB
	ARG [13]	Argentina	[3.90, 4.29]	RGB
	Nagoya [15]	Japan	[7.36, 11.84]	RGB
	Tokyo [15]	Japan	[15.73, 22.77]	RGB
	Vaihingen [80]	Germany	[2.36, 3.57]	NIR, G, B

(a)

Real-World Land Cover Mapping Datasets

Types	Datasets	Region	Categories
Target Domain	OEM [101]	Global	8
	Vaihingen [80]	Germany	6
	Potsdam [80]	Germany	6
	JAX [45]	US	6
	OMA [45]	US	6

(b)

5.1 Source-only Scenarios

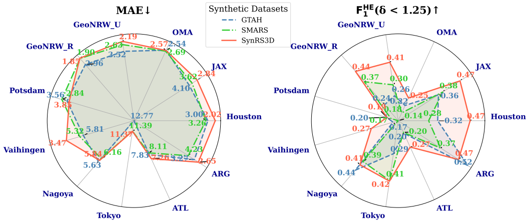


Figure 5: Source-only height estimation comparison of SynRS3D and other synthetic datasets showing in different metrics.

Height Estimation. We compared SynRS3D with other synthetic datasets in a source-only height estimation scenario. As shown in Figure 5, SynRS3D outperformed competitors SMARS [76] and GTAH [104] in 9 out of 11 real datasets. This superiority is attributed to the smaller domain gap and diversity of SynRS3D, as well as the precise calculation of heights within 3D software. In contrast, heights in SMARS and GTAH are not normalized, requiring additional algorithms for normalization, which introduces inherent noise in their height values.

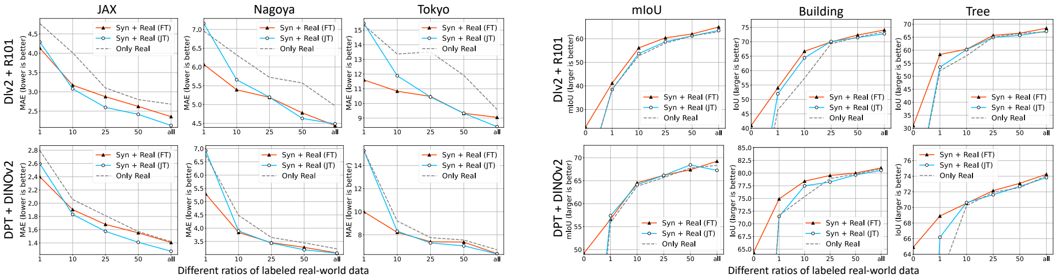
Land Cover Mapping. We demonstrate the source-only capability of models trained on SynRS3D in the land cover mapping task. Table 3 compares SynRS3D with existing synthetic datasets, SMARS [76] and SyntheWorld [86]. Due to category inconsistencies, we present IoU and mIoU for shared categories including trees, buildings, and ground on the JAX [45], OMA [45], Vaihingen [80], and Potsdam [80] datasets. The model trained on SynRS3D achieves the best results across these four real datasets using only land cover labels, demonstrating the extraordinary compatibility of SynRS3D.

Table 3: Source-only land cover mapping performance on various real-world datasets.

Datasets	JAX [45]				OMA [45]				Vaihingen [80]				Potsdam [80]			
	Ground	Tree	Building	mIoU	Ground	Tree	Building	mIoU	Ground	Tree	Building	mIoU	Ground	Tree	Building	mIoU
SMARS [76]	76.02	43.13	61.28	60.14	82.17	17.25	59.94	53.12	74.10	58.40	74.35	68.95	68.56	5.35	57.51	43.81
SyntheWorld [86]	74.63	54.74	64.18	64.52	81.29	**45.83**	56.56	61.23	72.69	68.09	75.67	72.15	69.09	32.49	55.88	52.49
SynRS3D	**77.69**	**57.03**	**68.96**	**67.89**	**83.96**	41.08	**62.28**	**62.44**	**75.66**	**68.58**	**79.61**	**74.61**	**74.26**	**35.34**	**69.46**	**59.69**

5.2 Combining SynRS3D with Real Data Scenarios

An important use of synthetic data is to augment real-world data. To demonstrate this capability of SynRS3D, we conducted two experiments. First, we trained models on SynRS3D and fine-tuned them on real data. Second, we combined SynRS3D with real data for joint training. We experimented with two different backbones: DINOv2 [71]+DPT [74] and DeepLabV2 [11]+ResNet101 [29]. Figure 6 (a) showcases SynRS3D's performance in the height estimation task on three city datasets: JAX [45],



(a) Height Estimation.

(b) Land Cover Mapping.

Figure 6: Performance evaluation of combining SynRS3D with real data in (a) height estimation and (b) land cover mapping. Height estimation is evaluated on various real datasets, and land cover mapping is evaluated on OEM dataset, showing IoU for building, tree, and mIoU. FT: fine-tuning on real data after pre-training on SynRS3D, JT: joint training with SynRS3D and real data.

Nagoya [15], and Tokyo [15]. The results indicate that both approaches yield significant improvements when real data is scarce, with benefits diminishing as more real data is added. Additionally, the stronger the backbone, the smaller the improvement provided by SynRS3D, and vice versa. Figure 6 (b) illustrates SynRS3D's augmentation capability in the land cover mapping task on OEM [101] dataset, showing similar conclusions to the height estimation task.

5.3 Transfer SynRS3D to Real-World Scenarios

Table 4: Results of RS3DAda height estimation branch using DINOv2 [71] and DPT [74]. The experimental results are divided as follows: "Whole" denotes the evaluation results for the entire image. "High" signifies the experimental results for image regions above 3 meters. T.D.1 and T.D.2 correspond to *Target Domain 1* and *Target Domain 2*, respectively, as specified in Table 2. Avg. stands for the average value.

Model	MAE ↓		RMSE ↓		Accuracy Metrics [18] ↑			F_1^{HE} ↑		
Avg. T.D.1	Whole	High	Whole	High	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
Train-on-T.D.1	**1.272**	**3.363**	**2.381**	**4.329**	**0.379**	**0.463**	**0.510**	**0.617**	**0.710**	**0.742**
Source Only	2.557	5.617	4.128	6.705	0.123	0.192	0.246	0.372	0.491	0.552
RS3DAda	2.148	4.921	3.593	6.024	0.185	0.258	0.318	0.418	0.554	0.623
Avg. T.D.2	Whole	High	Whole	High	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
Train-on-T.D.1	5.378	8.302	8.301	10.714	0.146	0.244	0.336	0.384	0.535	0.627
Source Only	6.117	8.923	9.221	11.443	0.125	0.223	0.312	0.365	0.514	0.601
RS3DAda	**4.866**	**7.227**	**7.584**	**9.594**	**0.182**	**0.299**	**0.389**	**0.485**	**0.621**	**0.689**

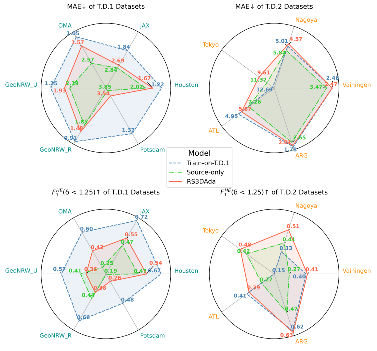


Figure 7: Results of RS3DAda height estimation branch for each dataset.

Heigh Estimation Branch. Table 4 shows the height estimation results for our RS3DAda model. "Whole" refers to the entire image, and "High" focuses on targets above 3 meters, such as trees and buildings. Using DINOv2 [71] and DPT [74] models, RS3DAda reduces the average MAE by 0.409 meters over the source-only approach across six datasets in *Target Domain 1*, though it is still exceeded by the models trained directly on these datasets. In *Target Domain 2*, RS3DAda outperforms models trained on real-world data, indicating its strong generalization under challenging scenarios, featuring diverse geographic regions and complex terrain characteristics. Figure 7 aligns with these results, showing RS3DAda's improvements on each dataset in *Target Domain 1* and *Target Domain 2*. This demonstrates SynRS3D's potential and the effectiveness of RS3DAda. However, the gap in *Target Domain 1* highlights the ongoing need to bridge the synthetic-to-real data gap, providing a benchmark for future UDA algorithm development in height estimation tasks.

Land Cover Mapping Branch. Table 5 presents the results of the RS3DAda model for the land cover mapping branch, evaluated using the OEM dataset. As shown, the RS3DAda method surpasses DAFormer by 1.94 in mIoU, indicating that the height branch positively impacts the land cover mapping performance. However, there remains a gap of 20.11 in mIoU compared to the Oracle

Table 5: Results of the RS3DAda land cover mapping branch on the OEM [101] dataset. All models are implemented with DINOv2 [71] and DPT [74].

Model	Bareland	Rangeland	Developed	Road	Tree	Water	Agriculture	Buildings	**mIoU**
Source-only	8.69	37.95	**22.54**	**49.05**	60.16	46.64	35.40	**65.19**	40.70
DAFormer [35]	12.54	41.16	10.88	43.88	**62.56**	**77.55**	62.62	59.10	46.29
RS3DAda	**19.92**	**47.61**	18.41	44.06	61.04	71.66	**63.73**	59.42	**48.23**
Train-on-OEM	50.04	59.10	58.18	65.39	73.07	83.65	76.36	80.88	68.34

model, suggesting significant room for improvement. Future research can build upon our method to make further advancements.

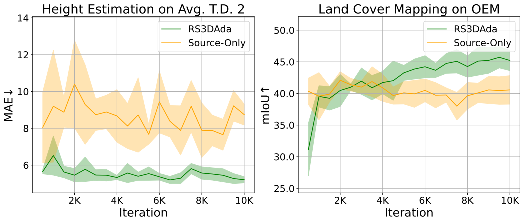


Figure 8: Performance of SynRS3D at the beginning of training for height estimation and land cover mapping branches, with and without the use of RS3DAda.

Stabilizing Training on SynRS3D. RS3DAda can regularize the training of synthetic data to prevent rapid overfitting at the beginning of the training, corresponding to the green section in Figure 8. Without RS3DAda, the model's evaluation results on the target domain fluctuate wildly during training in both height estimation and land cover mapping branches. This instability can lead to unreliable performance and poor generalization. RS3DAda ensures more consistent training, resulting in better model accuracy and stability.

Table 6: Comparison of RS3DAda with existing UDA methods AdaSeg and DADA. Supervision types: H for height maps, L for land cover labels. T.D.1 represents Target Domain 1, and T.D.2 represents Target Domain 2. V: Vaihingen [80]; P: Potsdam [80]; J: JAX [45]; O: OMA [45]. All models are implemented with DeepLabv2 and ResNet101. The datasets used for Tranin-on-Real are T.D.1 and OEM respectively.

Model	Supervision	Height Estimation (MAE) ↓		Land Cover Mapping (mIoU) ↑	
		Avg. T.D.1	Avg. T.D.2	OEM [101]	Avg. (V+P+J+O)
Source-Only	H, L	3.911	7.419	17.42	39.61
AdaptSeg [91]	L	-	-	20.06	40.00
DADA [94]	H+L	3.615	6.997	21.24	46.44
RS3DAda	H+L	**3.275**	**6.708**	**22.55**	**47.28**
Train-on-Real	H, L	1.859	6.639	64.54	53.12

Comparison with Existing UDA. We re-implemented AdaptSeg [91], DADA [94] and RS3DAda using DeepLabv2 [11] and ResNet101 [29] for fair comparison. Table 6 shows that RS3DAda outperformed both AdaptSeg and DADA in height estimation and land cover mapping tasks. Notably, when using weaker network architectures, the CNN encoder pre-trained on ImageNet fails to provide reliable RS image features and high accuracy for the ground category. As a result, the Feature Constraint and Ground-Guided Pseudo-Label Refinement in our RS3DAda cannot achieve their maximum effectiveness.

6 Conclusion and Discussion

In this work, we introduced SynRS3D, the largest synthetic remote sensing (RS) dataset, and RS3DAda, a multi-task unsupervised domain adaptation (UDA) method, designed to address the challenge of global 3D semantic reconstruction from single-view RS images. Our experiments on public datasets demonstrate the effectiveness of these tools in enhancing the use of synthetic data for RS research, setting a benchmark for 3D reconstruction from monocular RS images.

While SynRS3D offers a substantial contribution, there remains an appearance gap between synthetic and real-world data, potentially affecting real-world performance. Additionally, the dataset, though extensive, does not capture the full diversity of global cities, which could limit its generalizability. Future work will focus on reducing this gap and expanding the dataset's coverage to improve its robustness across different urban environments. By making these resources publicly available, we hope to stimulate further research and development in the field.

Acknowledgments

This work was supported in part by JST FOREST Program Grant Number JPMJFR206S; Microsoft Research Asia; JSPS KAKENHI Grant Number 24KJ0652; the Next Generation AI Research Center of The University of Tokyo; the Japan Science and Technology Agency SPRING Program (JST SPRING) Grant Number JPMJSP2108; and RIKEN Junior Research Associate (JRA) Program.

References

[1] Alexey Abramov, Christopher Bayer, and Claudio Heller. Keep it simple: Image statistics matching for domain adaptation. *arXiv preprint arXiv:2005.12551*, 2020.

[2] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

[3] B Ameri, N Goldstein, H Wehn, A Moshkovitz, and H Zwick. High resolution digital surface model (dsm) generation using multi-view multi-frame digital airborne images. *INTERNATIONAL ARCHIVES OF PHOTOGRAMMETRY REMOTE SENSING AND SPATIAL INFORMATION SCIENCES*, 34(4):419–424, 2002.

[4] Amir Atapour-Abarghouei and Toby P Breckon. Real-time monocular depth estimation using synthetic data with domain adaptation via image style transfer. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2800–2810, 2018.

[5] Gerald Baier, Antonin Deschemps, Michael Schmitt, and Naoto Yokoya. Synthesizing optical and sar imagery from land cover maps and auxiliary raster data. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–12, 2021.

[6] Wele Gedara Chaminda Bandara and Vishal M Patel. A transformer-based siamese network for change detection. In *IGARSS 2022-2022 IEEE International Geoscience and Remote Sensing Symposium*, pages 207–210. IEEE, 2022.

[7] Nicolas Bourdis, Denis Marraud, and Hichem Sahbi. Constrained optical flow for aerial image change detection. In *2011 IEEE international geoscience and remote sensing symposium*, pages 4176–4179. IEEE, 2011.

[8] Daniel J Butler, Jonas Wulff, Garrett B Stanley, and Michael J Black. A naturalistic open source movie for optical flow evaluation. In *Computer Vision–ECCV 2012: 12th European Conference on Computer Vision, Florence, Italy, October 7-13, 2012, Proceedings, Part VI 12*, pages 611–625. Springer, 2012.

[9] Hao Chen and Zhenwei Shi. A spatial-temporal attention-based method and a new dataset for remote sensing image change detection. *Remote Sensing*, 12(10):1662, 2020.

[10] Hongruixuan Chen, Jian Song, Chengxi Han, Junshi Xia, and Naoto Yokoya. Changemamba: Remote sensing change detection with spatiotemporal state space model. *IEEE Transactions on Geoscience and Remote Sensing*, 62:1–20, 2024.

[11] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence*, 40(4):834–848, 2017.

[12] Yun-Chun Chen, Yen-Yu Lin, Ming-Hsuan Yang, and Jia-Bin Huang. Crdoco: Pixel-level domain transfer with cross-domain consistency. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1791–1800, 2019.

[13] Gordon Christie, Kevin Foster, Shea Hagstrom, Gregory D Hager, and Myron Z Brown. Single view geocentric pose in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1162–1171, 2021.

[14] Blender Online Community. *Blender - a 3D modelling and rendering package*. Blender Foundation, Stichting Blender Foundation, Amsterdam, 2018.

[15] NTT DATA Corporation and Inc. DigitalGlobe. Aw3d high-resolution dataset. End User License Agreement, 2018. Available from NTT DATA Corporation and DigitalGlobe, Inc.

[16] Ilke Demir, Krzysztof Koperski, David Lindenbaum, Guan Pang, Jing Huang, Saikat Basu, Forest Hughes, Devis Tuia, and Ramesh Raskar. Deepglobe 2018: A challenge to parse the earth through satellite images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 172–181, 2018.

[17] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[18] David Eigen, Christian Puhrsch, and Rob Fergus. Depth map prediction from a single image using a multi-scale deep network. *Advances in neural information processing systems*, 27, 2014.

[19] Massimiliano Favalli, Alessandro Fornaciai, Ilaria Isola, Simone Tarquini, and Luca Nannipieri. Multiview 3d reconstruction in geosciences. *Computers & Geosciences*, 44:168–176, 2012.

[20] Jian Gao, Jin Liu, and Shunping Ji. A general deep learning based framework for 3d reconstruction from multi-view stereo satellite images. *ISPRS Journal of Photogrammetry and Remote Sensing*, 195:446–461, 2023.

[21] Zhi Gao, Wenbo Sun, Yao Lu, Yichen Zhang, Weiwei Song, Yongjun Zhang, and Ruifang Zhai. Joint learning of semantic segmentation and height estimation for remote sensing image leveraging contrastive learning. *IEEE Transactions on Geoscience and Remote Sensing*, 2023.

[22] Pedram Ghamisi and Naoto Yokoya. Img2dsm: Height simulation from single imagery using conditional generative adversarial net. *IEEE Geoscience and Remote Sensing Letters*, 15(5):794–798, 2018.

[23] Rui Gong, Wen Li, Yuhua Chen, Dengxin Dai, and Luc Van Gool. Dlow: Domain flow and applications. *International Journal of Computer Vision*, 129(10):2865–2888, 2021.

[24] Ziyang Gong, Fuhao Li, Yupeng Deng, Deblina Bhattacharjee, Xiangwei Zhu, and Zhenming Ji. Coda: Instructive chain-of-domain adaptation with severity-aware visual prompt tuning. *arXiv preprint arXiv:2403.17369*, 2024.

[25] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems*, pages 2672–2680, 2014.

[26] C Gordon, R Rene Rai Munoz Abujder, K Foster, S Hagstrom, GD Hager, and MZ Brown. Learning geocentric object pose in oblique monocular images. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2020.

[27] Yilong Han, Shugen Wang, Danchao Gong, Yue Wang, and X Ma. State of the art in digital surface modelling from multi-view high-resolution satellite images. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 2:351–356, 2020.

[28] Ankur Handa, Viorica Pătrăucean, Simon Stent, and Roberto Cipolla. Scenenet: An annotated model generator for indoor scene understanding. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5737–5743. IEEE, 2016.

[29] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[30] Txomin Hermosilla, Luis A Ruiz, Jorge A Recio, and Javier Estornell. Evaluation of automatic building detection approaches combining high resolution images and lidar data. *Remote Sensing*, 3(6):1188–1210, 2011.

[31] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020.

[32] Judy Hoffman, Eric Tzeng, Taesung Park, Jun-Yan Zhu, Phillip Isola, Kate Saenko, Alexei Efros, and Trevor Darrell. Cycada: Cycle-consistent adversarial domain adaptation. In *International conference on machine learning*, pages 1989–1998. Pmlr, 2018.

[33] Judy Hoffman, Dequan Wang, Fisher Yu, and Trevor Darrell. Fcns in the wild: Pixel-level adversarial and constraint-based adaptation. *arXiv preprint arXiv:1612.02649*, 2016.

[34] Danfeng Hong, Bing Zhang, Hao Li, Yuxuan Li, Jing Yao, Chenyu Li, Martin Werner, Jocelyn Chanussot, Alexander Zipf, and Xiao Xiang Zhu. Cross-city matters: A multimodal remote sensing benchmark dataset for cross-city semantic segmentation using high-resolution domain adaptation networks. *Remote Sensing of Environment*, 299:113856, 2023.

[35] Lukas Hoyer, Dengxin Dai, and Luc Van Gool. Daformer: Improving network architectures and training strategies for domain-adaptive semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9924–9935, 2022.

[36] Lukas Hoyer, Dengxin Dai, and Luc Van Gool. Hrda: Context-aware high-resolution domain-adaptive semantic segmentation. In *European conference on computer vision*, pages 372–391. Springer, 2022.

[37] Lukas Hoyer, Dengxin Dai, Haoran Wang, and Luc Van Gool. Mic: Masked image consistency for context-enhanced domain adaptation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11721–11732, 2023.

[38] Zhihua Hu, Yaolin Hou, Pengjie Tao, and Jie Shan. Imgtr: Image-triangle based multi-view 3d reconstruction for urban scenes. *ISPRS Journal of Photogrammetry and Remote Sensing*, 181:191–204, 2021.

[39] Peter J Huber. Robust estimation of a location parameter. In *Breakthroughs in statistics: Methodology and distribution*, pages 492–518. Springer, 1992.

[40] Javed Iqbal and Mohsen Ali. Weakly-supervised domain adaptation for built-up region segmentation in aerial and satellite imagery. *ISPRS Journal of Photogrammetry and Remote Sensing*, 167:263–275, 2020.

[41] Shunping Ji, Shiqing Wei, and Meng Lu. Fully convolutional networks for multisource building extraction from an open aerial and satellite imagery data set. *IEEE Transactions on Geoscience and Remote Sensing*, 57(1):574–586, 2018.

[42] Joon-Seok Kim, Hamdi Kavak, and Andrew Crooks. Procedural city generation beyond game development. *SIGSPATIAL Special*, 10(2):34–41, 2018.

[43] Fanjie Kong, Bohao Huang, Kyle Bradbury, and Jordan Malof. The synthinel-1 dataset: A collection of high resolution synthetic overhead imagery for building segmentation. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 1814–1823, 2020.

[44] Saket Kunwar. U-net ensemble for semantic and height estimation using coarse-map initialization. In *IGARSS 2019-2019 IEEE International Geoscience and Remote Sensing Symposium*, pages 4959–4962. IEEE, 2019.

[45] Bertrand Le Saux, Naoto Yokoya, Ronny Hänsch, and Myron Brown. 2019 ieee grss data fusion contest: large-scale semantic 3d reconstruction. *IEEE Geoscience and Remote Sensing Magazine (GRSM)*, 7(4):33–36, 2019.

[46] Dong-Hyun Lee et al. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on challenges in representation learning, ICML*, volume 3, page 896. Atlanta, 2013.

[47] Matthew J Leotta, Chengjiang Long, Bastien Jacquet, Matthieu Zins, Dan Lipsa, Jie Shan, Bo Xu, Zhixin Li, Xu Zhang, Shih-Fu Chang, et al. Urban semantic 3d reconstruction from multiview satellite imagery. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019.

[48] Fuhao Li, Ziyang Gong, Yupeng Deng, Xianzheng Ma, Renrui Zhang, Zhenming Ji, Xiangwei Zhu, and Hong Zhang. Parsing all adverse scenes: Severity-aware semantic segmentation with mask-enhanced cross-domain consistency. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 13483–13491, 2024.

[49] Qingyu Li, Lichao Mou, Yuansheng Hua, Yilei Shi, Sining Chen, Yao Sun, and Xiao Xiang Zhu. 3dcentripetalnet: Building height retrieval from monocular remote sensing imagery. *International Journal of Applied Earth Observation and Geoinformation*, 120:103311, 2023.

[50] Suo Li, Zhanyu Zhu, Haipeng Wang, and Feng Xu. 3d virtual urban scene reconstruction from a single optical remote sensing image. *IEEE Access*, 7:68305–68315, 2019.

[51] Wang Li, Zheng Niu, Rong Shang, Yuchu Qin, Li Wang, and Hanyue Chen. High-resolution mapping of forest canopy height using machine learning by coupling icesat-2 lidar with sentinel-1, sentinel-2 and landsat-8 data. *International Journal of Applied Earth Observation and Geoinformation*, 92:102163, 2020.

[52] Weijia Li, Lingxuan Meng, Jinwang Wang, Conghui He, Gui-Song Xia, and Dahua Lin. 3d building reconstruction from monocular remote sensing images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12548–12557, 2021.

[53] Weijia Li, Haote Yang, Zhenghao Hu, Juepeng Zheng, Gui-Song Xia, and Conghui He. 3d building reconstruction from monocular remote sensing images with multi-level supervisions. *arXiv preprint arXiv:2404.04823*, 2024.

[54] Xiang Li, Mingyang Wang, and Yi Fang. Height estimation from single aerial images using a deep ordinal regression network. *IEEE Geoscience and Remote Sensing Letters*, 19:1–5, 2020.

[55] Xin Li, Feng Xu, Fan Liu, Xin Lyu, Yao Tong, Zhennan Xu, and Jun Zhou. A synergistical attention model for semantic segmentation of remote sensing images. *IEEE Transactions on Geoscience and Remote Sensing*, 61:1–16, 2023.

[56] Qing Lian, Fengmao Lv, Lixin Duan, and Boqing Gong. Constructing self-motivated pyramid curriculums for cross-domain semantic segmentation: A non-adversarial approach. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6758–6767, 2019.

[57] Jin Liu, Jian Gao, Shunping Ji, Chang Zeng, Shaoyi Zhang, and JianYa Gong. Deep learning based multi-view stereo matching and 3d scene reconstruction from oblique aerial images. *ISPRS Journal of Photogrammetry and Remote Sensing*, 204:42–60, 2023.

[58] Weixing Liu, Jun Liu, Xin Su, Han Nie, and Bin Luo. Source-free domain adaptive object detection in remote sensing images. *arXiv preprint arXiv:2401.17916*, 2024.

[59] Yi Liu, Chao Pang, Zongqian Zhan, Xiaomeng Zhang, and Xue Yang. Building change detection for remote sensing images using a dual-task constrained deep siamese convolutional network model. *IEEE Geoscience and Remote Sensing Letters*, 18(5):811–815, 2020.

[60] Yuheng Liu, Yifan Zhang, Ye Wang, and Shaohui Mei. Rethinking transformers for semantic segmentation of remote sensing images. *IEEE Transactions on Geoscience and Remote Sensing*, 2023.

[61] Adrian Lopez-Rodriguez and Krystian Mikolajczyk. Desc: Domain adaptation for depth estimation via semantic consistency. *International Journal of Computer Vision*, 131(3):752–771, 2023.

[62] Xiaoqiang Lu, Tengfei Gong, and Xiangtao Zheng. Multisource compensation network for remote sensing cross-domain scene classification. *IEEE Transactions on Geoscience and Remote Sensing*, 58(4):2504–2515, 2019.

[63] Yawei Luo, Liang Zheng, Tao Guan, Junqing Yu, and Yi Yang. Taking a closer look at domain shift: Category-level adversaries for semantics consistent domain adaptation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2507–2516, 2019.

[64] A Mahphood, H Arefi, A Hosseininaveh, and AA Naeini. Dense multi-view image matching for dsm generation from satellite images. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 42:709–715, 2019.

[65] Yongqiang Mao, Kaiqiang Chen, Liangjin Zhao, Wei Chen, Deke Tang, Wenjie Liu, Zhirui Wang, Wenhui Diao, Xian Sun, and Kun Fu. Elevation estimation-driven building 3d reconstruction from single-view remote sensing imagery. *IEEE Transactions on Geoscience and Remote Sensing*, 2023.

[66] Nikolaus Mayer, Eddy Ilg, Philip Hausser, Philipp Fischer, Daniel Cremers, Alexey Dosovitskiy, and Thomas Brox. A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4040–4048, 2016.

[67] Ke Mei, Chuang Zhu, Jiaqi Zou, and Shanghang Zhang. Instance adaptive self-training for unsupervised domain adaptation. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXVI 16*, pages 415–430. Springer, 2020.

[68] Pascal Müller, Peter Wonka, Simon Haegler, Andreas Ulmer, and Luc Van Gool. Procedural modeling of buildings. In *ACM SIGGRAPH 2006 Papers*, pages 614–623. 2006.

[69] F Kenton Musgrave, Craig E Kolb, and Robert S Mace. The synthesis and rendering of eroded fractal terrains. *ACM Siggraph Computer Graphics*, 23(3):41–50, 1989.

[70] Viktor Olsson, Wilhelm Tranheden, Juliano Pinto, and Lennart Svensson. Classmix: Segmentation-based data augmentation for semi-supervised learning. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 1369–1378, 2021.

[71] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

[72] Esam Othman, Yakoub Bazi, Farid Melgani, Haikel Alhichri, Naif Alajlan, and Mansour Zuair. Domain adaptation network for cross-scene classification. *IEEE Transactions on Geoscience and Remote Sensing*, 55(8):4441–4456, 2017.

[73] Claudio Persello, Ronny Hänsch, Gemine Vivone, Kaiqiang Chen, Zhiyuan Yan, Deke Tang, Hai Huang, Michael Schmitt, and Xian Sun. 2023 ieee grss data fusion contest: Large-scale fine-grained building classification for semantic urban reconstruction [technical committees]. *IEEE Geoscience and Remote Sensing Magazine*, 11(1):94–97, 2023.

[74] René Ranftl, Alexey Bochkovskiy, and Vladlen Koltun. Vision transformers for dense prediction. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 12179–12188, 2021.

[75] Mario Fuentes Reyes, Pablo d'Angelo, and Friedrich Fraundorfer. Syntcities: A large synthetic remote sensing dataset for disparity estimation. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 15:10087–10098, 2022.

[76] Mario Fuentes Reyes, Yuxing Xie, Xiangtian Yuan, Pablo d'Angelo, Franz Kurz, Daniele Cerra, and Jiaojiao Tian. A 2d/3d multimodal data simulation approach with applications on urban semantic segmentation, building extraction and change detection. *ISPRS Journal of Photogrammetry and Remote Sensing*, 205:74–97, 2023.

[77] Stephan R Richter, Vibhav Vineet, Stefan Roth, and Vladlen Koltun. Playing for data: Ground truth from computer games. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part II 14*, pages 102–118. Springer, 2016.

[78] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022.

[79] German Ros, Laura Sellart, Joanna Materzynska, David Vazquez, and Antonio M Lopez. The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3234–3243, 2016.

[80] Franz Rottensteiner, Gunho Sohn, Jaewook Jung, Markus Gerke, Caroline Baillard, Sebastien Benitez, and Uwe Breitkopf. The isprs benchmark on urban object classification and 3d building reconstruction. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences; I-3*, 1(1):293–298, 2012.

[81] Ewelina Rupnik, Marc Pierrot-Deseilligny, and Arthur Delorme. 3d reconstruction from multi-view vhr-satellite images in micmac. *ISPRS Journal of Photogrammetry and Remote Sensing*, 139:201–211, 2018.

[82] Christos Sakaridis, Dengxin Dai, Simon Hecker, and Luc Van Gool. Model adaptation with synthetic and real data for semantic dense foggy scene understanding. In *Proceedings of the european conference on computer vision (ECCV)*, pages 687–704, 2018.

[83] Michael Schmitt, Seyed Ali Ahmadi, Yonghao Xu, Gülşen Taşkın, Ujjwal Verma, Francescopaolo Sica, and Ronny Hänsch. There are no data like more data: Datasets for deep learning in earth observation. *IEEE Geoscience and Remote Sensing Magazine*, 2023.

[84] Jacob Shermeyer, Thomas Hossler, Adam Van Etten, Daniel Hogan, Ryan Lewis, and Daeil Kim. Rareplanes: Synthetic data takes flight. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 207–217, 2021.

[85] Gunho Sohn and Ian J Dowman. Extraction of buildings from high resolution satellite data and lidar. In *XX ISPRS CONGRESS*, 2004.

[86] Jian Song, Hongruixuan Chen, and Naoto Yokoya. Syntheworld: A large-scale synthetic dataset for land cover mapping and building change detection. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 8287–8296, 2024.

[87] Shivangi Srivastava, Michele Volpi, and Devis Tuia. Joint height estimation and semantic labeling of monocular aerial images with cnns. In *2017 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pages 5173–5176. IEEE, 2017.

[88] Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. *Advances in neural information processing systems*, 30, 2017.

[89] Onur Tasar, Yuliya Tarabalka, Alain Giros, Pierre Alliez, and Sébastien Clerc. Standardgan: Multi-source domain adaptation for semantic segmentation of very high resolution satellite images by data standardization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 192–193, 2020.

[90] Xin-Yi Tong, Gui-Song Xia, Qikai Lu, Huanfeng Shen, Shengyang Li, Shucheng You, and Liangpei Zhang. Land-cover classification with high-resolution remote sensing images using transferable deep models. *Remote Sensing of Environment*, 237:111322, 2020.

[91] Yi-Hsuan Tsai, Wei-Chih Hung, Samuel Schulter, Kihyuk Sohn, Ming-Hsuan Yang, and Manmohan Chandraker. Learning to adapt structured output space for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7472–7481, 2018.

[92] Yi-Hsuan Tsai, Kihyuk Sohn, Samuel Schulter, and Manmohan Chandraker. Domain adaptation for structured output via discriminative patch representations. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1456–1465, 2019.

[93] Tuan-Hung Vu, Himalaya Jain, Maxime Bucher, Matthieu Cord, and Patrick Pérez. Advent: Adversarial entropy minimization for domain adaptation in semantic segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2517–2526, 2019.

[94] Tuan-Hung Vu, Himalaya Jain, Maxime Bucher, Matthieu Cord, and Patrick Perez. Dada: Depth-aware domain adaptation in semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019.

[95] Haoran Wang, Tong Shen, Wei Zhang, Ling-Yu Duan, and Tao Mei. Classes matter: A fine-grained adversarial approach to cross-domain semantic segmentation. In *European conference on computer vision*, pages 642–659. Springer, 2020.

[96] Junjue Wang, Zhuo Zheng, Ailong Ma, Xiaoyan Lu, and Yanfei Zhong. Loveda: A remote sensing land-cover dataset for domain adaptive semantic segmentation. *arXiv preprint arXiv:2110.08733*, 2021.

[97] Xiaolei Wang, Zirong Hu, Shouhai Shi, Mei Hou, Lei Xu, and Xiang Zhang. A deep learning method for optimizing semantic segmentation accuracy of remote sensing images based on improved unet. *Scientific reports*, 13(1):7600, 2023.

[98] Ximei Wang, Ying Jin, Mingsheng Long, Jianmin Wang, and Michael I Jordan. Transferable normalization: Towards improving transferability of deep neural networks. *Advances in neural information processing systems*, 32, 2019.

[99] World Bank. World development indicators 2024, 2024.

[100] Magnus Wrenninge and Jonas Unger. Synscapes: A photorealistic synthetic dataset for street scene parsing. *arXiv preprint arXiv:1810.08705*, 2018.

[101] Junshi Xia, Naoto Yokoya, Bruno Adriano, and Clifford Broni-Bediako. Openearthmap: A benchmark dataset for global high-resolution land cover mapping. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 6254–6264, 2023.

[102] Ruijie Xiao, Chuan Zhong, Wankang Zeng, Ming Cheng, and Cheng Wang. Novel convolutions for semantic segmentation of remote sensing images. *IEEE Transactions on Geoscience and Remote Sensing*, 2023.

[103] Yuxing Xie, Xiangtian Yuan, Xiao Xiang Zhu, and Jiaojiao Tian. Multimodal co-learning for building change detection: A domain adaptation framework using vhr images and digital surface models. *IEEE Transactions on Geoscience and Remote Sensing*, 2024.

[104] Zhitong Xiong, Wei Huang, Jingtao Hu, and Xiao Xiang Zhu. The benchmark: Transferable representation learning for monocular height estimation. *IEEE Transactions on Geoscience and Remote Sensing*, 2023.

[105] Yang Xu, Bohao Huang, Xiong Luo, Kyle Bradbury, and Jordan M Malof. Simpl: Generating synthetic overhead imagery to address custom zero-shot and few-shot detection problems. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 15:4386–4396, 2022.

[106] Yonghao Xu, Bo Du, Liangpei Zhang, Daniele Cerra, Miguel Pato, Emiliano Carmona, Saurabh Prasad, Naoto Yokoya, Ronny Hänsch, and Bertrand Le Saux. Advanced multi-sensor optical remote sensing for urban land use and land cover classification: Outcome of the 2018 ieee grss data fusion contest. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 12(6):1709–1724, 2019.

[107] Jiayu Yang, Jose M Alvarez, and Miaomiao Liu. Self-supervised learning of depth inference for multi-view stereo. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7526–7534, 2021.

[108] Kunping Yang, Gui-Song Xia, Zicheng Liu, Bo Du, Wen Yang, Marcello Pelillo, and Liangpei Zhang. Semantic change detection with asymmetric siamese networks. *arXiv preprint arXiv:2010.05687*, 2020.

[109] Lihe Yang, Bingyi Kang, Zilong Huang, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything: Unleashing the power of large-scale unlabeled data. *arXiv preprint arXiv:2401.10891*, 2024.

[110] Yanchao Yang and Stefano Soatto. Fda: Fourier domain adaptation for semantic segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4085–4095, 2020.

[111] Yu-Ting Yen, Chia-Ni Lu, Wei-Chen Chiu, and Yi-Hsuan Tsai. 3d-pl: Domain adaptive depth estimation with 3d-aware pseudo-labeling. In *European Conference on Computer Vision*, pages 710–728. Springer, 2022.

[112] Dawen Yu, Shunping Ji, Jin Liu, and Shiqing Wei. Automatic 3d building reconstruction from multi-view aerial images with deep learning. *ISPRS Journal of Photogrammetry and Remote Sensing*, 171:155–170, 2021.

[113] Dawen Yu, Shunping Ji, Jin Liu, and Shiqing Wei. Automatic 3d building reconstruction from multi-view aerial images with deep learning. *ISPRS Journal of Photogrammetry and Remote Sensing*, 171:155–170, 2021.

[114] Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6023–6032, 2019.

[115] Qiming Zhang, Jing Zhang, Wei Liu, and Dacheng Tao. Category anchor-guided unsupervised domain adaptation for semantic segmentation. *Advances in neural information processing systems*, 32, 2019.

[116] Zuxun Zhang, Jun Wu, Yong Zhang, Yongjun Zhang, and Jianqing Zhang. Multi-view 3d city model generation with image sequences. *INTERNATIONAL ARCHIVES OF PHOTOGRAMMETRY REMOTE SENSING AND SPATIAL INFORMATION SCIENCES*, 34(5/W12):351–356, 2003.

[117] Shanshan Zhao, Huan Fu, Mingming Gong, and Dacheng Tao. Geometry-aware symmetric domain adaptation for monocular depth estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9788–9798, 2019.

[118] Wufan Zhao, Claudio Persello, and Alfred Stein. Semantic-aware unsupervised domain adaptation for height estimation from single-view aerial images. *ISPRS Journal of Photogrammetry and Remote Sensing*, 196:372–385, 2023.

[119] Chuanxia Zheng, Tat-Jen Cham, and Jianfei Cai. T2net: Synthetic-to-realistic translation for solving single-image depth estimation tasks. In *Proceedings of the European conference on computer vision (ECCV)*, pages 767–783, 2018.

[120] Zhuo Zheng, Yanfei Zhong, and Junjue Wang. Pop-net: Encoder-dual decoder for semantic segmentation and single-view height estimation. In *IGARSS 2019-2019 IEEE International Geoscience and Remote Sensing Symposium*, pages 4963–4966. IEEE, 2019.

[121] Zheng Zhou, Change Zheng, Xiaodong Liu, Ye Tian, Xiaoyi Chen, Xuexue Chen, and Zixun Dong. A dynamic effective class balanced approach for remote sensing imagery semantic segmentation of imbalanced data. *Remote Sensing*, 15(7):1768, 2023.

[122] Yang Zou, Zhiding Yu, BVK Kumar, and Jinsong Wang. Unsupervised domain adaptation for semantic segmentation via class-balanced self-training. In *Proceedings of the European conference on computer vision (ECCV)*, pages 289–305, 2018.

[123] Yang Zou, Zhiding Yu, Xiaofeng Liu, BVK Kumar, and Jinsong Wang. Confidence regularized self-training. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 5982–5991, 2019.

[124] Zhengxia Zou, Tianyang Shi, Wenyuan Li, Zhou Zhang, and Zhenwei Shi. Do game data generalize well for remote sensing image segmentation? *Remote Sensing*, 12(2):275, 2020.

Appendix

A Technical Supplements

In this technical supplement, we provide detailed insights and additional results to support our main paper. Section A.1 outlines the generation process of the SynRS3D dataset, including the tools and plugins used. It also covers the licenses for these plugins. Section A.2 discusses the data sources and licenses of the existing real-world datasets utilized in our experiments. Section A.3 elaborates on the evaluation metrics for different tasks, including the proposed F_1^{HE} metric specifically designed for remote sensing height estimation tasks. Section A.4 describes the experimental setup and the selection of hyperparameters for the RS3DAda method. Section A.5 presents the ablation study results and analysis for the RS3DAda method. Section A.6 provides supplementary experimental results combining SynRS3D and real data scenarios, complementing Section 5.2 of the main paper. Section A.7 showcases the qualitative visual results of RS3DAda on various tasks. Section A.8 details the generation process and samples of building change detection annotations in SynRS3D, as well as the evaluation results of the source-only scenario on different real datasets. Section A.9 highlights the performance of models trained on the SynRS3D dataset using RS3DAda in the critical application of disaster mapping in remote sensing.

A.1 Detailed Generation Workflow of SynRS3D

The generation workflow of SynRS3D involves several key steps, from initializing sensor and sunlight parameters to generating the layout, geometry, and textures of the scene. This comprehensive process ensures that the generated SynRS3D mimics real-world remote sensing scenarios with high fidelity.

The main steps of the workflow are as follows:

- **Initialization:** Set up the sensor and sunlight parameters using uniform and normal distributions to simulate various conditions.
- **Layout Generation:** Define the grid and terrain parameters to create diverse urban and natural environments.
- **Geometry Generation:** Specify the characteristics of roads, rivers, buildings, and vegetation, ensuring realistic representations.
- **Texture Generation:** Use advanced models like GPT-4 [2] and Stable Diffusion [78] to generate realistic textures for different categories of land cover.
- **Scene Construction and Processing:** Assemble the scene with all generated components and apply textures to create visually accurate post-event and pre-event images.
- **Outlier Filtering:** Filter outliers based on height maps to ensure the quality and reliability of the dataset.

The detailed algorithm for this workflow is provided in Algorithm 1. The development process of SynRS3D is based on Blender 3.4, where we utilized and modified various community add-ons to facilitate the generation of SynRS3D. A comprehensive list of all the add-ons used during our development process is presented in Table 7.

Table 7: List of Blender add-ons used in the SynRS3D.

Name	Author	Version	License	URL
Realtime River Generator	specoolar	1.1	RF	`https://blendermarket.com/products/river-generator`
Next Street	Next Realm	2.0	RF	`https://blendermarket.com/products/next-street`
Objects Replacer	Georeality Design	1.06	GPL	`https://blendermarket.com/products/objects-replacer/docs`
Albero	Greenbaburu	0.3	RF	`https://blendermarket.com/products/albero---geometry-nodes-powered-tree-generator`
Hira Building Generator	HiranojiStore	0.9	RF	`https://blendermarket.com/products/hira-building-generator`
Procedural Building Generator	Isak Waltin	1.2.1	CC-BY 4.0	`https://blendermarket.com/products/building-gen`
Pro Atmo	Contrastrender	1.0	GPL	`https://blendermarket.com/products/pro-atmo`
Modular Buildings Creator	PH Felix	1.0	RF	`https://blendermarket.com/products/modular-buildings-creator`
Next Trees	Next Realm	2.0	RF	`https://blendermarket.com/products/next-trees`
SceneCity	Arnaud	1.9.3	RF	`http://www.cgchan.com/store/scenecity`
Flex Road Generator	EasyNodes	1.1.0	RF	`https://www.cgtrader.com/3d-models/scripts-plugins/modelling/blender-mesh-curve-to-road`
Buildify	Pavel Oliva	1.0	RF	`https://paveloliva.gumroad.com/l/buildify`

A.2 License and Data Source of Real-World Datasets

The licenses and data sources for the real-world datasets used for evaluation and training in this work are shown in Table 8. For the Potsdam, Vaihingen, GeoNRW, Nagoya, and Tokyo datasets, we used the dsm2dtm [2] algorithm to convert them to normalized Digital Surface Model (nDSM), since they only provide Digital Surface

[2] https://github.com/seedlit/dsm2dtm

Algorithm 1 Generation Workflow of SynRS3D

```
Initialize Parameters
$\mathcal{S} \leftarrow \{\text{azimuth} \sim U(a_1, a_2), \text{look\_angle} \sim \mathcal{N}(\mu_1, \sigma_1), \text{GSD} \sim \mathcal{N}(\mu_2, \sigma_2)\}$ # $\mathcal{S}$: Sensor parameters
$\mathcal{L} \leftarrow \{\text{elevation} \sim U(e_1, e_2), \text{intensity} \sim U(i_1, i_2), \text{color} \sim [U(c_1, c_2), U(c_1, c_2), U(c_1, c_2)]\}$ # $\mathcal{L}$: Sunlight parameters
Generate Layout
$\mathcal{G} \leftarrow \{\text{district\_num} \sim \text{randint}(d_1, d_2), \text{district\_size} \sim \text{randint}(s_1, s_2), \text{obj\_density} \sim U(o_1, o_2)\}$ # $\mathcal{G}$: Grid parameters
$\mathcal{T} \leftarrow \{\text{flat\_area} \sim U(f_1, f_2), \text{mountain\_area} \sim U(m_1, m_2), \text{sea\_area} \sim U(s_1, s_2), \text{tree\_density} \sim U(t_1, t_2)\}$ # $\mathcal{T}$: Terrain parameters
Generate Geometry
$\mathcal{R} \leftarrow \{\text{river\_num} \sim \text{randint}(r_1, r_2), \text{road\_num} \sim \text{randint}(r_3, r_4), \text{width} \sim U(w_1, w_2)\}$ # $\mathcal{R}$: Road and River parameters
$\mathcal{B} \leftarrow \{\text{height} \sim U(h_1, h_2), \text{type} \in \text{select}(\text{types}), \text{roof\_angle} \sim U(ra_1, ra_2)\}$ # $\mathcal{B}$: Building parameters
$\mathcal{V} \leftarrow \{\text{trunk} \sim \text{Sample\_Curve}(), \text{branch\_num} \sim \text{randint}(b_1, b_2), \text{leaf\_num} \sim \text{randint}(l_1, l_2)\}$ # $\mathcal{V}$: Tree parameters
Generate Textures
$\mathcal{C} \leftarrow \{\text{Rangeland}, \text{Agricultural Land}, \text{Bareland}, \text{Developed Space}, \text{Road}, \text{Roof}\}$ # $\mathcal{C}$: Texture categories
for category $\in \mathcal{C}$ do
    texture_prompts $\leftarrow$ GPT-4(category)
    textures[category] $\leftarrow$ Stable_Diffusion(texture_prompts)
end for
Construct Scene
$\mathcal{P}_s \leftarrow \text{create\_scene}(\mathcal{S} \cup \mathcal{L} \cup \mathcal{G} \cup \mathcal{T} \cup \mathcal{R} \cup \mathcal{B} \cup \mathcal{V})$ # $\mathcal{P}_s$: Post-event scene
$\mathcal{Q}_s \leftarrow \text{remove\_buildings}(\text{copy}(\mathcal{P}_s), U(rb_1, rb_2))$ # $\mathcal{Q}_s$: Pre-event scene
Process Scene
$\mathcal{P}_t \leftarrow \text{apply\_textures}(\mathcal{P}_s, \text{textures})$ # $\mathcal{P}_t$: Post-event scene with textures
$\mathcal{Q}_t \leftarrow \text{apply\_textures}(\mathcal{Q}_s, \text{textures})$ # $\mathcal{Q}_t$: Pre-event scene with textures
$\mathcal{P}_r \leftarrow \text{render\_rgb}(\mathcal{P}_t)$ # $\mathcal{P}_r$: Post-event RGB image
$\mathcal{Q}_r \leftarrow \text{render\_rgb}(\mathcal{Q}_t)$ # $\mathcal{Q}_r$: Pre-event RGB image
$\mathcal{P}_l \leftarrow \text{generate\_land\_cover}(\mathcal{P}_t)$ # $\mathcal{P}_l$: Post-event land cover mapping
$\mathcal{P}_h \leftarrow \text{generate\_height\_map}(\mathcal{P}_t)$ # $\mathcal{P}_h$: Post-event height map
$\mathcal{P}_b \leftarrow \text{generate\_building\_mask}(\mathcal{P}_t)$ # $\mathcal{P}_b$: Post-event building mask
$\mathcal{Q}_b \leftarrow \text{generate\_building\_mask}(\mathcal{Q}_t)$ # $\mathcal{Q}_b$: Pre-event building mask
$\mathfrak{C} \leftarrow \text{subtract\_masks}(\mathcal{P}_b, \mathcal{Q}_b)$ # $\mathfrak{C}$: Building change detection mask
Filter Outliers # Input: $\mathcal{P}_h$, $H_T$, $H_m$, $H_s$; Output: $\mathcal{F}_{\mathcal{P}_h}$ (Filtered height map list)
$H_T \leftarrow$ threshold value # Set the height threshold value
$H_m \leftarrow$ minimum threshold # Set the minimum proportion threshold
$H_s \leftarrow$ steepness value # Set the steepness value for the sigmoid function
$\mathcal{F}_{\mathcal{P}_h} \leftarrow \emptyset$ # Initialize the filtered height map set
for each $n \in \mathcal{P}_h$ do
    $a \leftarrow \text{read\_image}(n)$ # Read the height map as a numpy array
    $T_p \leftarrow \text{total\_pixels}(a)$ # Calculate the total number of pixels
    $A_t \leftarrow \text{count\_above\_threshold}(a, H_T)$ # Count the number of pixels above the threshold
    $P_c \leftarrow \frac{A_t}{T_p}$ # Calculate the proportion of pixels above the threshold
    if $P_c \geq H_m$ then
        $\mathcal{F}_{\mathcal{P}_h} \leftarrow \mathcal{F}_{\mathcal{P}_h} \cup \{n\}$ # If proportion is above minimum threshold, add to filtered list
    else
        $Pr \leftarrow \frac{1}{1+e^{-H_s \cdot (P_c - H_m)}}$ # Calculate the probability using a sigmoid function
        if random() $< Pr$ then
            $\mathcal{F}_{\mathcal{P}_h} \leftarrow \mathcal{F}_{\mathcal{P}_h} \cup \{n\}$ # Add to filtered list based on probability
        end if
    end if
end for
Output # Output: SynRS3D dataset
$\{\mathcal{F}_{\mathcal{P}_r}, \mathcal{F}_{\mathcal{Q}_r}, \mathcal{F}_{\mathcal{P}_l}, \mathcal{F}_{\mathcal{P}_h}, \mathcal{F}_{\mathfrak{C}}\}$ # $\mathcal{F}_{\mathcal{P}_r}$: Filtered post-event RGB images, $\mathcal{F}_{\mathcal{Q}_r}$: Filtered pre-event RGB images, $\mathcal{F}_{\mathcal{P}_l}$: Filtered post-event land cover mappings, $\mathcal{F}_{\mathcal{P}_h}$: Filtered post-event height maps, $\mathcal{F}_{\mathfrak{C}}$: Filtered building change detection masks
```

Table 8: The data source and license of real-world height estimation datasets used in this work.

Real-World Datasets			
Types	**Datasets**	**Data Source**	**License/Conditions of Use**
Target Domain 1	Houston [106]	Data Fusion Contest 2018	Creative Commons Attribution
	JAX [45]	Data Fusion Contest 2019	Creative Commons Attribution
	OMA [45]	Data Fusion Contest 2019	Creative Commons Attribution
	GeoNRW_Urban [5]	GeoNRW	Creative Commons Attribution
	GeoNRW_Rural [5]	GeoNRW	Creative Commons Attribution
	Potsdam [80]	ISPRS	Research Purposes Only, No Redistribution
Target Domain 2	ATL [13]	Overhead Geopose Challenge	Creative Commons Attribution
	ARG [13]	Overhead Geopose Challenge	Creative Commons Attribution
	Nagoya [15]	NTT DATA Corporation and Inc. DigitalGlobe	End User License Agreement
	Tokyo [15]	NTT DATA Corporation and Inc. DigitalGlobe	End User License Agreement
	Vaihingen [80]	ISPRS	Research Purposes Only, No Redistribution

Model (DSM). We will release the processed real-world datasets upon acceptance, provided that the original datasets are allowed to be redistributed and are intended for non-commercial use.

A.3 Evaluation Metrics

We utilized several metrics to ensure a comprehensive assessment of model performance when evaluating land cover mapping and height estimation tasks. In the following parts, we provide a detailed explanation and formulation of adopted metrics.

A.3.1 Land Cover Mapping

Intersection over Union (IoU) Intersection over Union (IoU) is a common evaluation metric used in image segmentation tasks. It measures the overlap between the predicted segmentation and the ground truth segmentation. The IoU for a single class is defined as:

$$\text{IoU} = \frac{|A \cap B|}{|A \cup B|}, \tag{6}$$

where A is the set of predicted pixels and B is the set of ground truth pixels.

Mean Intersection over Union (mIoU) mIoU extends IoU to multiple classes by averaging the IoU values of all classes. If there are N classes, mIoU is calculated as:

$$\text{mIoU} = \frac{1}{N} \sum_{i=1}^{N} \text{IoU}_i, \tag{7}$$

where IoU_i is the IoU for class i. This metric provides a single scalar value that summarizes the segmentation performance across all classes.

A.3.2 Height Estimation

Mean Absolute Error (MAE) Mean Absolute Error (MAE) measures the average magnitude of the errors between the predicted heights and the true heights. Suppose the ground truth heights are Y and the predicted heights are $\hat{Y}$, and n is the number of samples. It is defined as:

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^{n} |Y_i - \hat{Y}_i|. \tag{8}$$

Root Mean Squared Error (RMSE) Root Mean Squared Error (RMSE) measures the square root of the average squared differences between predicted heights and actual heights. It is defined as:

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (Y_i - \hat{Y}_i)^2}. \tag{9}$$

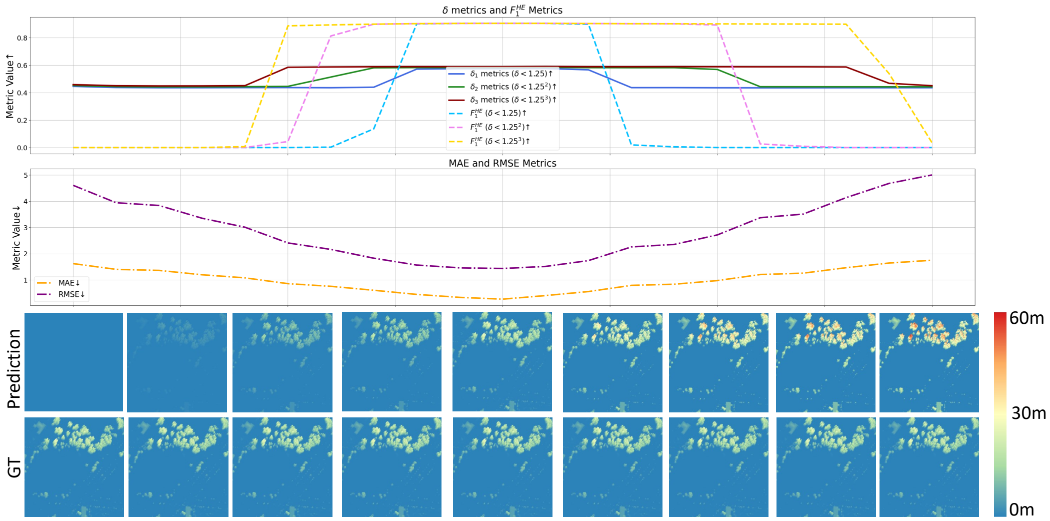


Figure 9: Comparison of proposed F_1^{HE} metric and other metrics.

Accuracy Metric This metric, also called δ metric from early depth estimation work [18], evaluates the proportion of height predictions that fall within a certain ratio of the true heights. We use δ to represent a maxRatio map, which is calculated as follows:

$$\delta = \max\left(\frac{\hat{Y}}{Y}, \frac{Y}{\hat{Y}}\right). \tag{10}$$

Then, threshold values η are used to measure the accuracy of the height predictions, the values of η are usually $1.25, 1.25^2, 1.25^3$.

F1 Score for Height Estimation (F_1^{HE}) The F_1^{HE} score innovatively applies the F1 score, typically used in classification, to the regression task of height estimation. This metric emphasizes both precision and recall in estimating significant heights. The F_1^{HE} score balances precision and recall for height predictions above a significance threshold T (e.g., 1 meter). The maxRatio is calculated as in equation 10. True Positives (TP), False Positives (FP), and False Negatives (FN) are identified as follows:

$$TP = \sum\left((\hat{Y} > T \wedge Y > T) \wedge (\delta < \eta)\right), \tag{11}$$

$$FP = \sum\left(\hat{Y} > T \wedge Y \leq T\right), \tag{12}$$

$$FN = \sum\left(\hat{Y} \leq T \wedge Y > T\right), \tag{13}$$

where the values of η are usually $1.25, 1.25^2, 1.25^3$. Precision, Recall, and F_1^{HE} are then calculated as:

$$Precision = \frac{TP}{TP + FP}, \tag{14}$$

$$Recall = \frac{TP}{TP + FN}, \tag{15}$$

$$F_1^{HE} = 2 \times \frac{Precision \times Recall}{Precision + Recall}. \tag{16}$$

Our motivation for proposing a new metric for height estimation arises from observing that existing metrics such as MAE, RMSE, and δ metrics, which are derived from depth estimation tasks, do not consider the unique characteristics of height estimation in remote sensing images. Specifically, a significant portion of the remote sensing images can be occupied by ground classes, leading to an abundance of zero height values in the ground truth. This imbalance impedes the evaluation of model performance when using traditional depth estimation metrics.

Table 9: Settings for RS3DAda experimental hyperparameters.

Category	Parameter	Value
Statistical Image Translation [1]	Fourier Domain Adaptation (FDA)	beta_limit= 0.01
	Histogram Matching (HM)	blend_ratio= $[0.8, 1.0]$
	Pixel Distribution Adaptation (PDA)	blend_ratio= $[0.8, 1.0]$, transform_type="standard"
Strong Augmentation	ClassMix	$\lvert\mathcal{C}\rvert/2$
	ColorJitter [2]	$p = 0.8$
	GaussianBlur [3]	$p = 0.5$
Pseudo Label Generation	Land Cover Confidence Threshold (τ)	0.95
	Height Map Consistency Threshold (η)	1.55
Optimization	Optimizer	AdamW
	Encoder Learning Rate (lr)	1×10^{-6}
	Decoder Learning Rate	$10 \times lr$
	Weight Decay	5×10^{-4}
	Batch Size	2
	Iterations	$40,000$
	Warmup Steps	$1,500$
	Warmup Mode	Linear
	Decay Mode	Polynomial
	EMA (α)	0.99
Loss Function	Feature Loss Threshold (ϵ)	0.8
	Weighting Coefficient for Target Loss (λ_{target})	1
	Weighting Coefficient for Feature Loss (λ_{feat})	1

[1] https://albumentations.ai/docs/api_reference/augmentations/domain_adaptation/

[2] https://albumentations.ai/docs/api_reference/augmentations/transforms/#albumentations.augmentations.transforms.ColorJitter

[3] https://albumentations.ai/docs/api_reference/augmentations/blur/transforms/#albumentations.augmentations.blur.transforms.GaussianBlur

As illustrated in Figure 9, when a network predicts all values as 0 meters or predicts the height of trees and buildings as twice their ground truth (30 meters to 60 meters), metrics like MAE, RMSE, and δ still indicate highly competitive accuracy. This is not reasonable because these metrics average the correct predictions of a large number of ground pixels. However, in height estimation tasks, the accuracy of predictions for objects with height is crucial. Our proposed F_1^{HE} metric specifically addresses this issue by focusing on the accuracy of height predictions for objects higher than 1 meter. As shown, in both extreme cases, the F1 score is 0, reflecting the poor performance correctly. This metric better aligns with the objectives of the height estimation task. In practice, most images in height estimation datasets contain objects with heights exceeding 1 meter, so we skip the F_1^{HE} calculation for images that only contain ground pixels.

This comprehensive evaluation framework ensures that height estimation models are assessed on both overall error rates and the ability to accurately predict significant height values in remote sensing images.

A.4 Experimental Setting for RS3DAda

For the real-world datasets used in our experiments, we split each dataset into a 3:1 ratio for training and testing. In the RS3DAda experiments, we use random cropping of size 392 to ensure the dimensions are multiples of 14. The training batch size is set to 2, with each batch consisting of one labeled synthetic image from SynRS3D and one unlabeled image from the target domain training set.

Additionally, in RS3DAda, the teacher model is updated using Exponential Moving Average (EMA) of the student model parameters as follows:

$$\theta_t \leftarrow \alpha\theta_t + (1 - \alpha)\theta_s, \tag{17}$$

where θ_t represents the teacher model parameters, θ_s represents the student model parameters, and α is the EMA decay factor.

For detailed experimental parameters, please refer to Table 9.

A.5 Ablation Studies of RS3DAda

In this section, we mainly conduct ablation experiments on the three key modules of RS3DAda: 1) the ground mask, 2) height map consistency, and 3) feature constraints. Additionally, we performed ablation studies on different mixing strategies in the strong augmentation of the target domain and the setting of the number of categories in the land cover branch. The evaluation dataset for height estimation experiments is *Target Domain 2*. For land cover mapping experiments, we employed the OEM [101] dataset for evaluation.

Table 10 presents the ablation study results for the RS3DAda method. Specifically, using DINOv2 [71] and DPT [74], we find that all three modules are important for height estimation, with the ground mask and height

Table 10: Ablation experiments of two types of network structures with our key modules, which were introduced in the RS3DADa section. MAE and F_1^{HE} serve as evaluation metrics for the height estimation tasks, and IoU is used for the land cover mapping tasks.

#	Model	Ground Mask	Height Consistency	Feature Constraint	Height Estimation		Land Cover Mapping
					MAE ↓	F_1^{HE} ($\delta < 1.25$) ↑	mIoU ↑
1	DPT+DINOv2	–	–	–	6.117	0.365	42.60
2	DPT+DINOv2	✓	–	–	5.652	0.423	44.05
3	DPT+DINOv2	✓	✓	–	5.253	0.425	44.75
4	DPT+DINOv2	✓	–	✓	5.578	0.439	42.93
5	DPT+DINOv2	–	✓	✓	5.384	0.461	46.67
6	DPT+DINOv2	✓	✓	✓	**4.886**	**0.485**	**48.23**
7	DLv2+R101	–	–	–	7.419	0.318	17.42
8	DLv2+R101	✓	✓	✓	6.959	0.316	18.89
9	DLv2+R101	✓	✓	–	**6.708**	**0.352**	**22.55**

consistency being particularly crucial. For instance, in Experiments 1 and 2, adding the ground mask reduces MAE from 6.117 to 5.652 and increases F_1^{HE} from 0.365 to 0.423. Adding height consistency in Experiment 3 further improves performance, reducing MAE to 5.253 and increasing F_1^{HE} to 0.425. The feature constraint, shown in Experiment 4, also contributes to improvements, though its impact is less significant. When all three modules are used together in Experiment 6, the best results are achieved with a MAE of 4.886, F_1^{HE} of 0.485, and mIoU of 48.23. For land cover mapping, height consistency is essential. Without it, the model relies on land cover confidence for height regression, which is often insufficient. This lack of confidence in the pseudo labels for the height branch hinders the improvement of the height estimation branch, subsequently affecting the land cover branch. These results indicate that both branches support each other, and inadequate learning in one branch negatively impacts the other.

Interestingly, with the weaker network combination of DeepLabv2 [11] and ResNet101 [29] (Experiments 7-9), the feature constraint is ineffective. This is because the ImageNet-pretrained feature extractor, trained on natural images, does not generalize well to synthetic remote sensing data, unlike DINOv2's self-supervised pretraining on diverse datasets. Aligning features with the ImageNet-pretrained extractor hinders learning from synthetic data due to the significant domain gap. This demonstrates our method's effectiveness in leveraging DINOv2's features as a constraint.

Table 11: Comparison of mixing strategies and number of classes.

Mix Strategy / #Class	Height Estimation		Land Cover Mapping
	MAE ↓	F_1^{HE} ($\delta < 1.25$) ↑	mIoU ↑
Mix Strategy			
CutMix [114]	4.966	0.475	47.34
ClassMix [70]	**4.886**	**0.485**	**48.23**
#Classes			
3	5.136	0.425	-
8	**4.886**	**0.485**	-

We also explored the impact of two different mix strategies and the number of land cover classes on the RS3DAda method. As shown in Tab. 11, ClassMix has a slight advantage over CutMix in both tasks. Regarding the number of land cover classes, we found that using all 8 land cover classes outperforms using only 3 classes (ground, tree, building). This improvement is likely because land cover mapping, being a segmentation task, benefits from a more detailed and discrete representation of features. In contrast, height estimation, which is a regression task, relies on continuous features. By having a finer label space in the classification branch, we can better align the segmentation and regression tasks, reducing the discrepancy between them.

A.6 Additional Height Estimation Results in Combining SynRS3D and Real Data Scenarios

In the Section 5.2 of the main paper, we present height estimation results for three datasets. Here, we provide the remaining results for seven additional datasets. These results further demonstrate the efficacy of combining SynRS3D with real data across different environments for fine-tuning and joint training. Figures 10 and 11 showcase the performance across these additional datasets, following the same evaluation methodology as described in Section 5.2 of the main paper. These extended results support the main paper's conclusions, demonstrating that both fine-tuning on real data after pre-training on SynRS3D (FT) and joint training with

SynRS3D and real data (JT) significantly enhance model performance, especially when real data is limited. This underscores the importance of SynRS3D in complementing existing datasets and boosting model performance.

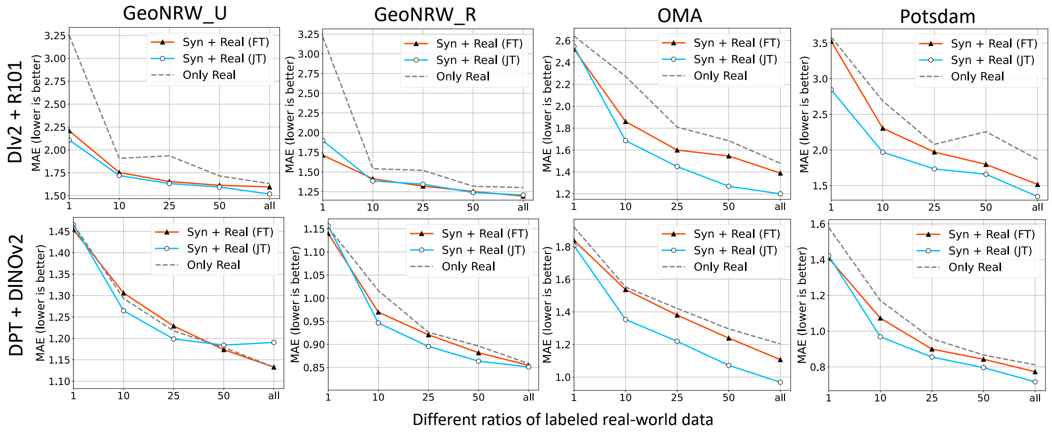


Figure 10: Additional performance evaluation on *Target Domain 1* datasets of combining SynRS3D with real data on height estimation task. FT: fine-tuning on real data after pre-training on SynRS3D, JT: joint training with SynRS3D and real data.

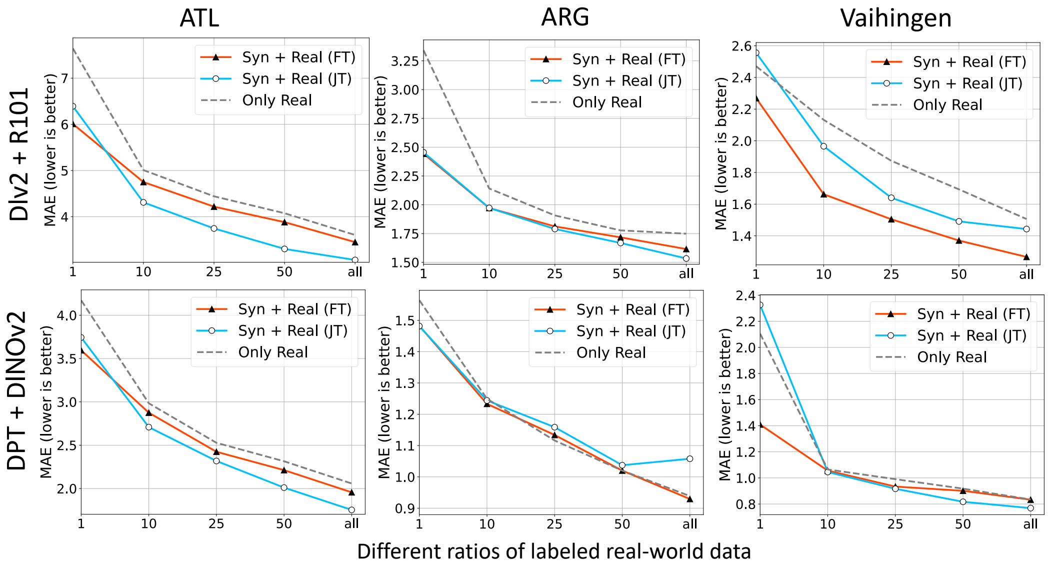


Figure 11: Additional performance evaluation on *Target Domain 2* datasets of combining SynRS3D with real data on height estimation task. FT: fine-tuning on real data after pre-training on SynRS3D, JT: joint training with SynRS3D and real data.

A.7 Qualitative Results of RS3DAda

Figure 12 shows the qualitative results for the height estimation task. We can observe that the height predictions from the RS3DAda model are closer to the ground truth and have more complete edges. In contrast, the source-only model tends to overestimate height values and produces more incomplete edges. Although the model trained on *Target Domain 1* uses real data, it struggles to generalize to *Target Domain 2* due to its training data being limited to commonly available public datasets from European and American regions, which are unbalanced. As shown, its predicted heights are often underestimated. Figure 13 presents the qualitative results for the land cover mapping task. The RS3DAda model demonstrates exceptional performance in categories such as agricultural land, rangeland, and bare land, which aligns with our quantitative experimental results. However, it has some limitations in categories like roads and developed space, indicating that there is still significant room for improvement in domain adaptation research for the SynRS3D dataset in the area of land cover mapping. This marks the first time in the field of remote sensing that synthetic data alone can achieve a high level of visual interpretation consistency with the ground truth. We hope that the RS3DAda method and the SynRS3D dataset can serve as benchmarks to further advance research in this direction. Figure 14 shows additional 3D reconstruction results in developing countries. These results are derived from using models trained on SynRS3D with RS3DAda to infer monocular satellite image tiles from Bing Satellite[3] and HereWeGo Satellite[4]. These 3D reconstruction areas cover between 3.2 square kilometers and 12.85 square kilometers, with a ground sample distance (GSD) of 0.35 meters.

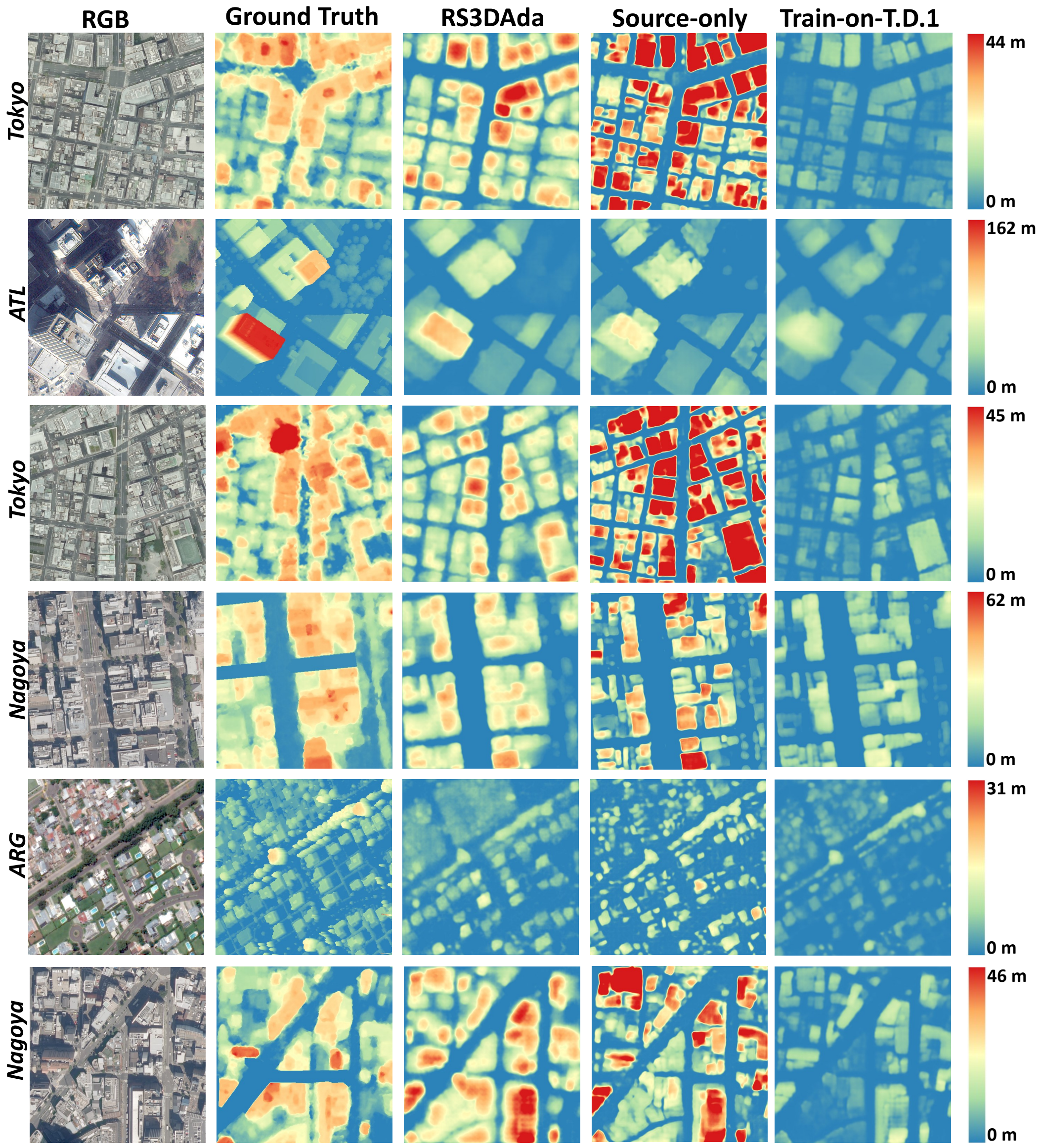


Figure 12: Qualitative results of height estimation task on *Target Domain 2* using the RS3DAda model, the source-only model, and the model trained on *Target Domain 1*. Satellite RGB images form Tokyo and Nagoya: © 2018 NTT DATA Corporation and Inc. DigitalGlobe.

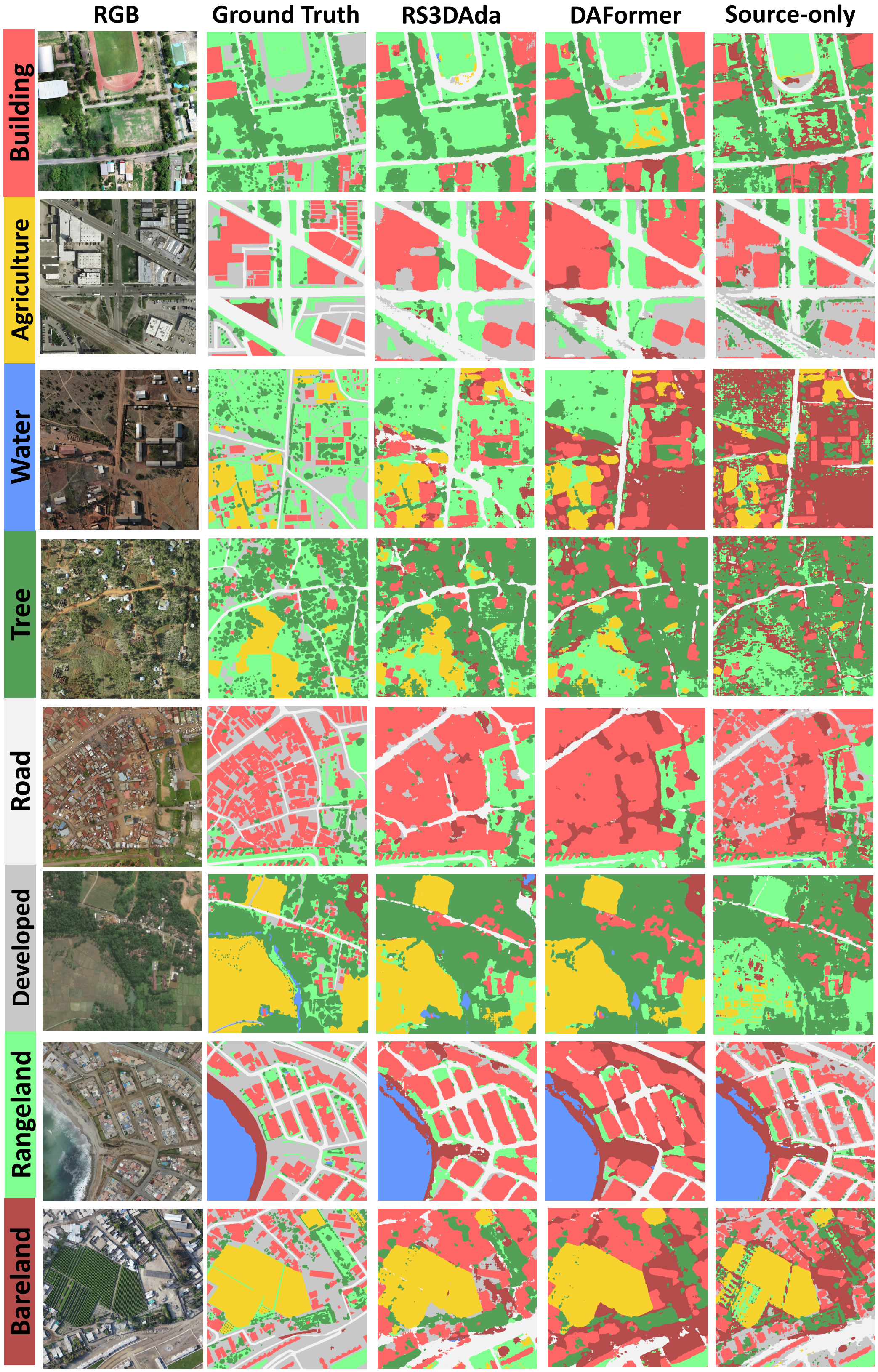


Figure 13: Qualitative results of land cover mapping task on OEM dataset using the RS3DAda model, the source-only model, and DAFormer.

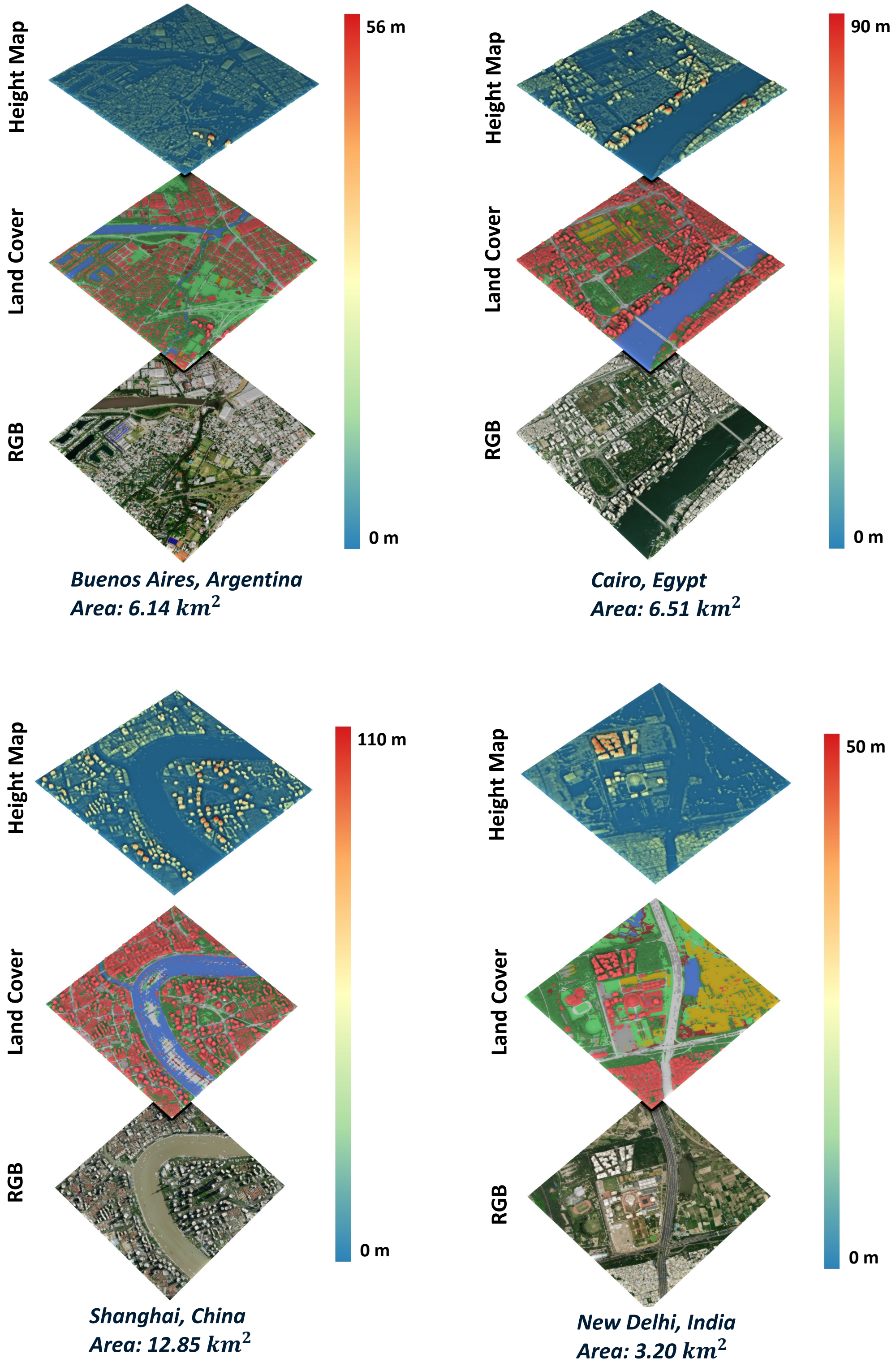


Figure 14: 3D visualization outcomes from real-world monocular RS images, which uses the model trained on SynRS3D dataset with proposed RS3DAda method. RGB satellite images of Buenos Aires and New Delhi: © HERE WeGo Satellite. RGB satellite images of Cairo and Shanghai: © Being Satellite.

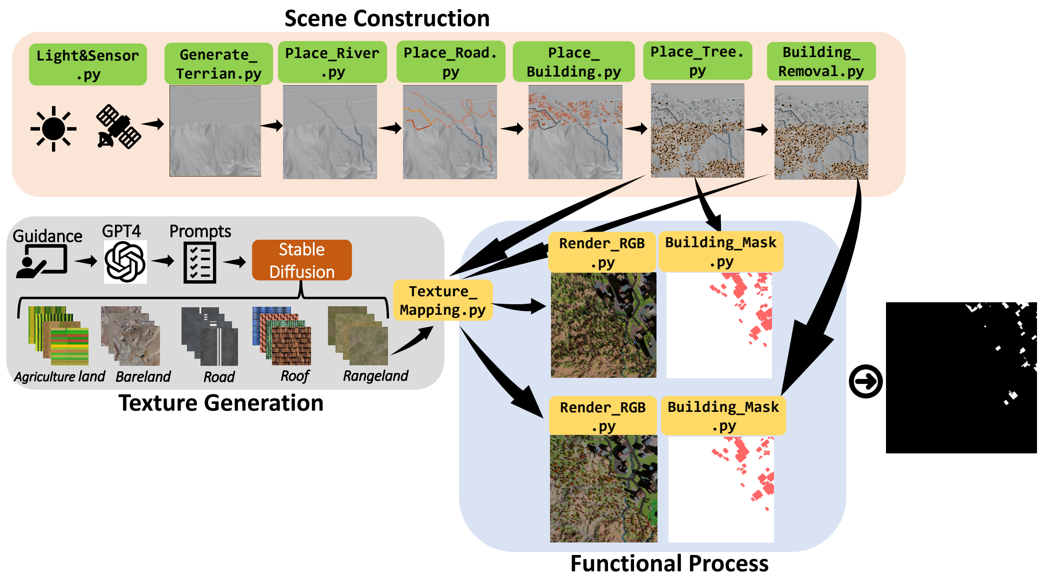


Figure 15: Generation workflow of building change detection mask.

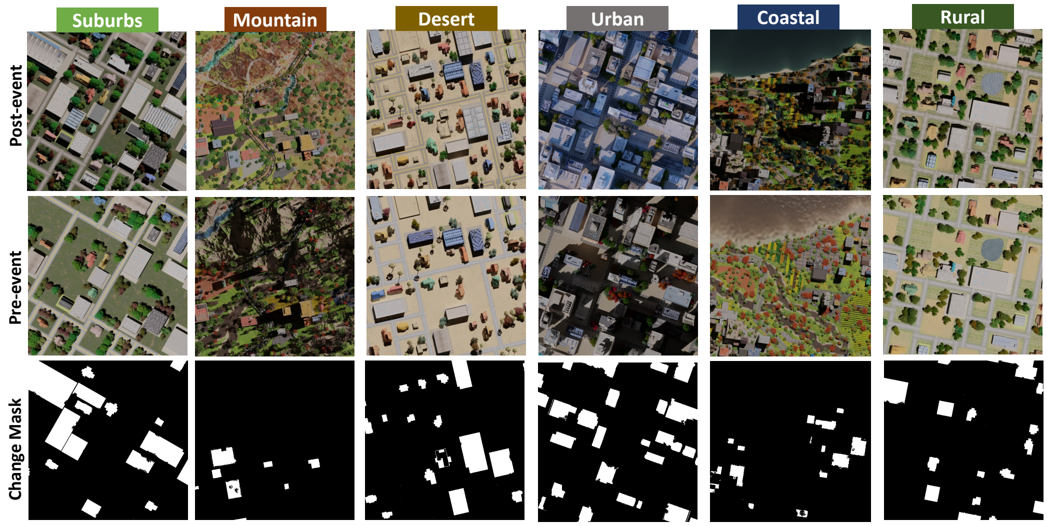


Figure 16: Examples of building change detection task in SynRS3D.

A.8 Building Change Detection

SynRS3D provides 8-class land cover mapping annotations, accurate height maps, and binary masks specifically designed for RS building change detection tasks. The image and mask generation process is illustrated in Figure 15. For the synthesized scenes, an additional step is included: a certain proportion of buildings are randomly removed, and all geometries in the scene are retextured. Subsequently, post-event and pre-event RGB images, along with building masks, are rendered. By subtracting the two masks, the final building change mask is obtained. Figure 16 exhibits examples of change detection in six styles within SynRS3D.

To validate the effectiveness of SynRS3D in change detection tasks, we conducted experiments in a source-only scenario, where models were trained only on synthetic data and tested directly on real-world datasets. We compared our results with the models trained on two other advanced synthetic datasets, SMARS [76] and SyntheWorld [86], that include labels for the RS building change detection task. For real-world datasets, we used commonly utilized datasets in change detection tasks: WHU-CD [41], LEVIR-CD+ [9], and SECOND [108].

[3] `https://www.bing.com/maps`

[4] `https://wego.here.com`

The WHU-CD dataset, a subset of the WHU Building dataset, focuses on building change detection with aerial images from Christchurch, New Zealand, captured in April 2012 and 2016 at 0.3 meters/pixel resolution. Covering 20.5 km^2, the dataset documents significant urban development, with buildings increasing from 12,796 to 16,077 over four years. LEVIR-CD+ is an advanced building CD dataset comprising 985 pairs of high-resolution (0.5 meters/pixel) images, documenting changes over 5 to 14 years and featuring various building types. It includes 31,333 instances of building changes, making it a valuable benchmark for CD methodologies. The SECOND dataset consists of 4,662 pairs of 512×512 aerial images (0.5-3 meters/pixel) annotated for land cover change detection in cities like Hangzhou, Chengdu, and Shanghai, but in our experiments, we only use its building change mask. These datasets were split into training and testing sets in a 3:1 ratio, with a training size of 256×256 pixels.

We employed four change detection frameworks for evaluating SMAR, SyntheWorld, and SynRS3D, including the CNN-based DTCDSCN [59], the transformer-based ChangeFormer [6], and the current state-of-the-art Mamba-based method, ChangeMamba [10]. Notably, due to our empirical findings of the strong potential of DINOV2 [71] pre-trained networks on synthetic data in both land cover mapping and change detection tasks, we implemented a framework combining the DINOV2 encoder with the ChangeMamba decoder for change detection on synthetic data, which we named DINOMamba. For synthetic datasets, we use a batch size of 2, and for real data, we use a batch size of 16. The optimizer used is AdamW, with a learning rate of 1e-5 for DinoMamba, while all other methods use a learning rate of 1e-4. All models are trained for 40,000 iterations on a single Tesla A100. The evaluation metrics used are IoU and F1.

Table 12: Peformance evaluation of building change detection task on WHU-CD [41] dataset.

Train on	DTCDSCN [59]		ChangeFormer [6]		ChangeMamba [10]		DinoMamba	
	IoU	F1	IoU	F1	IoU	F1	IoU	F1
SMARS [76]	26.84	42.55	18.67	31.88	42.50	59.63	48.11	64.87
SyntheWorld [86]	30.17	46.53	**41.73**	**58.87**	47.26	64.10	54.20	70.14
SynRS3D	**33.09**	**49.84**	35.00	51.94	**52.94**	**69.08**	**61.60**	**76.00**
Real	58.31	73.67	79.98	88.88	88.44	93.87	87.57	93.38

Table 13: Peformance evaluation of building change detection task on LEVIR-CD+ [9] dataset.

Train on	DTCDSCN [59]		ChangeFormer [6]		ChangeMamba [10]		DinoMamba	
	IoU	F1	IoU	F1	IoU	F1	IoU	F1
SMARS [76]	11.70	21.53	15.67	27.58	27.50	42.50	30.85	47.31
SyntheWorld [86]	21.16	35.28	23.31	38.12	28.28	44.30	48.78	65.46
SynRS3D	**25.82**	**41.30**	**23.33**	**38.14**	**30.39**	**46.78**	**49.63**	**66.23**
Real	63.44	77.63	67.48	80.58	77.39	87.25	74.12	85.14

Table 14: Peformance evaluation of building change detection task on the SECOND [108] dataset.

Train on	DTCDSCN [59]		ChangeFormer [6]		ChangeMamba [10]		DinoMamba	
	IoU	F1	IoU	F1	IoU	F1	IoU	F1
SMARS [76]	17.26	29.88	23.30	38.09	29.85	46.15	35.20	51.07
SyntheWorld [86]	21.00	35.07	26.44	42.06	27.23	43.02	37.61	54.71
SynRS3D	**33.52**	**50.32**	**31.36**	**47.90**	**38.88**	**56.02**	**39.18**	**56.33**
Real	58.78	74.04	60.08	75.06	67.61	80.68	67.65	80.71

Tables 12, 13, 14 present our experimental results, showing that the combination of SynRS3D and DINOMamba achieved F1 scores of 76.00, 66.23, and 56.33 on WHU, LEVIR-CD+, and SECOND respectively. Although there is still a gap compared to the Oracle model trained on real-world data, our dataset significantly boosts models' performances compared with the other two synthetic datasets. We have established a benchmark based on SynRS3D and advanced change detection networks, hoping to further promote the development of RS change detection using synthetic data.

A.9 Disaster Mapping Study Cases

The models trained on the SynRS3D dataset using the RS3DAda method can be utilized for various remote sensing downstream applications. We explored their potential in disaster mapping applications.

In February 2023, a devastating earthquake struck southeastern Turkey, primarily affecting the Kahramanmaraş region. This earthquake, with a magnitude of 7.8, caused widespread destruction, resulting in over 45,000 deaths, thousands of injuries, and massive displacement of residents. The economic losses were estimated to be in the billions of dollars. Rescue operations were carried out by both national and international teams, working tirelessly to save lives and provide aid to the affected population. Similarly, in August 2023, Hawaii experienced severe wildfires, particularly affecting the island of Maui. These wildfires, exacerbated by dry conditions and strong winds, led to extensive destruction of homes, infrastructure, and natural landscapes. The fires caused significant economic losses, displacing many residents and leading to casualties. The coordinated efforts of local authorities and fire departments, along with support from federal agencies, were crucial in controlling the fires and assisting those affected. To assess the impact of these disasters, we used the height estimation branch of RS3DAda to infer pre- and post-event remote sensing images. By simply subtracting the predicted height maps of the post-event from the pre-event, we obtained a Height Difference map. This map was filtered using a threshold: 3 meters for the earthquake example (indicating that buildings severely damaged in the earthquake would collapse, resulting in a significant height reduction) and 1 meter for the wildfire example (assuming that changes exceeding 1 meter indicate damage in the fire). Figure 18 presents the study case for the Turkey earthquake, and Figure 17 shows the study case for the Hawaii wildfires.

This simple method allowed us to roughly delineate the affected areas and assess the damage severity based on height differences. Although not entirely precise, this approach represents a significant success in applying models trained solely on synthetic data to real-world scenarios. We believe in the potential of RS3DAda and SynRS3D in this research domain and look forward to more applications and studies in the future.

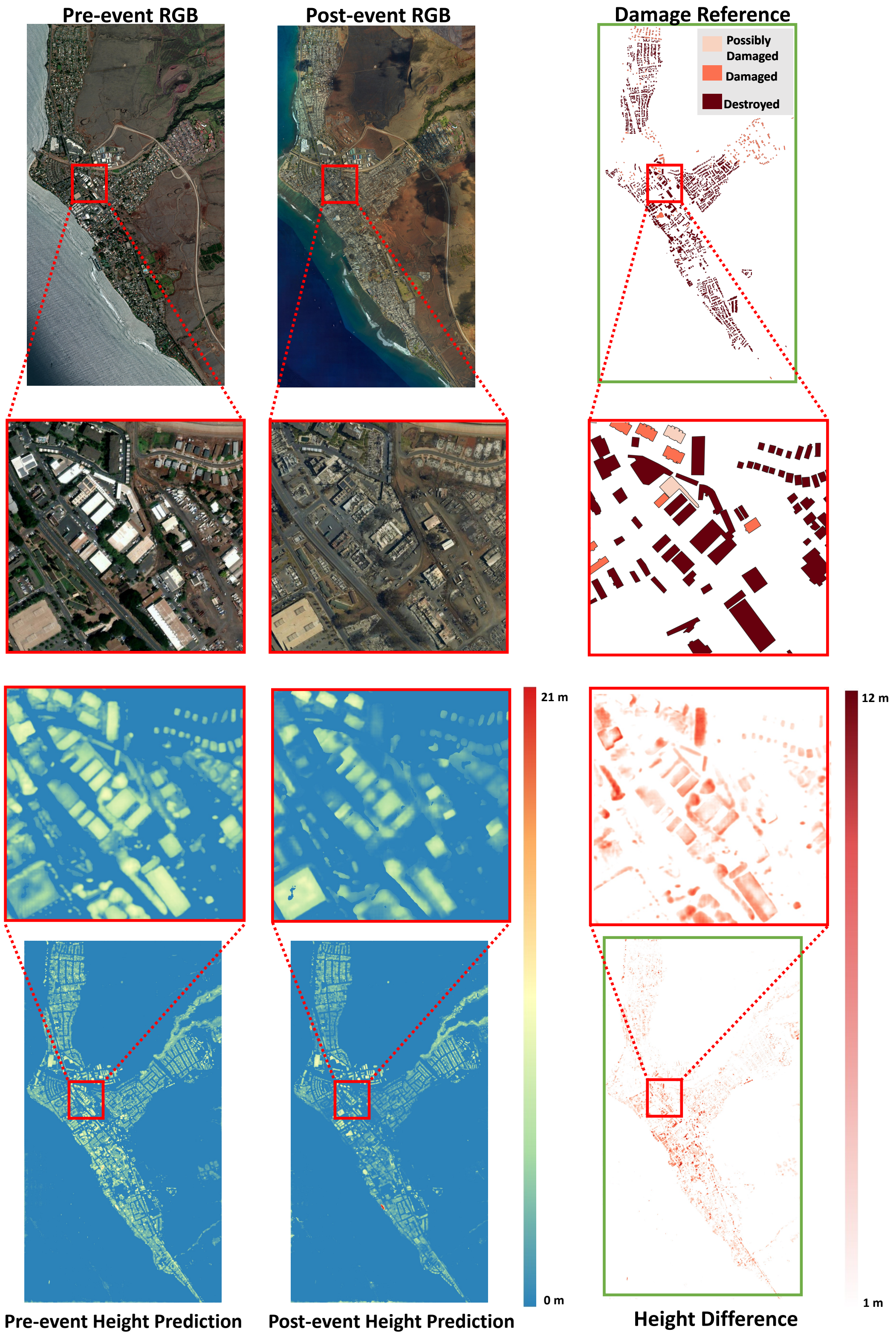


Figure 17: Study case of 2023 Hawaii-Maui wildfire. RGB satellite images of pre-event: © Being Satellite. RGB satellite images of post-event: © Google Satellite.

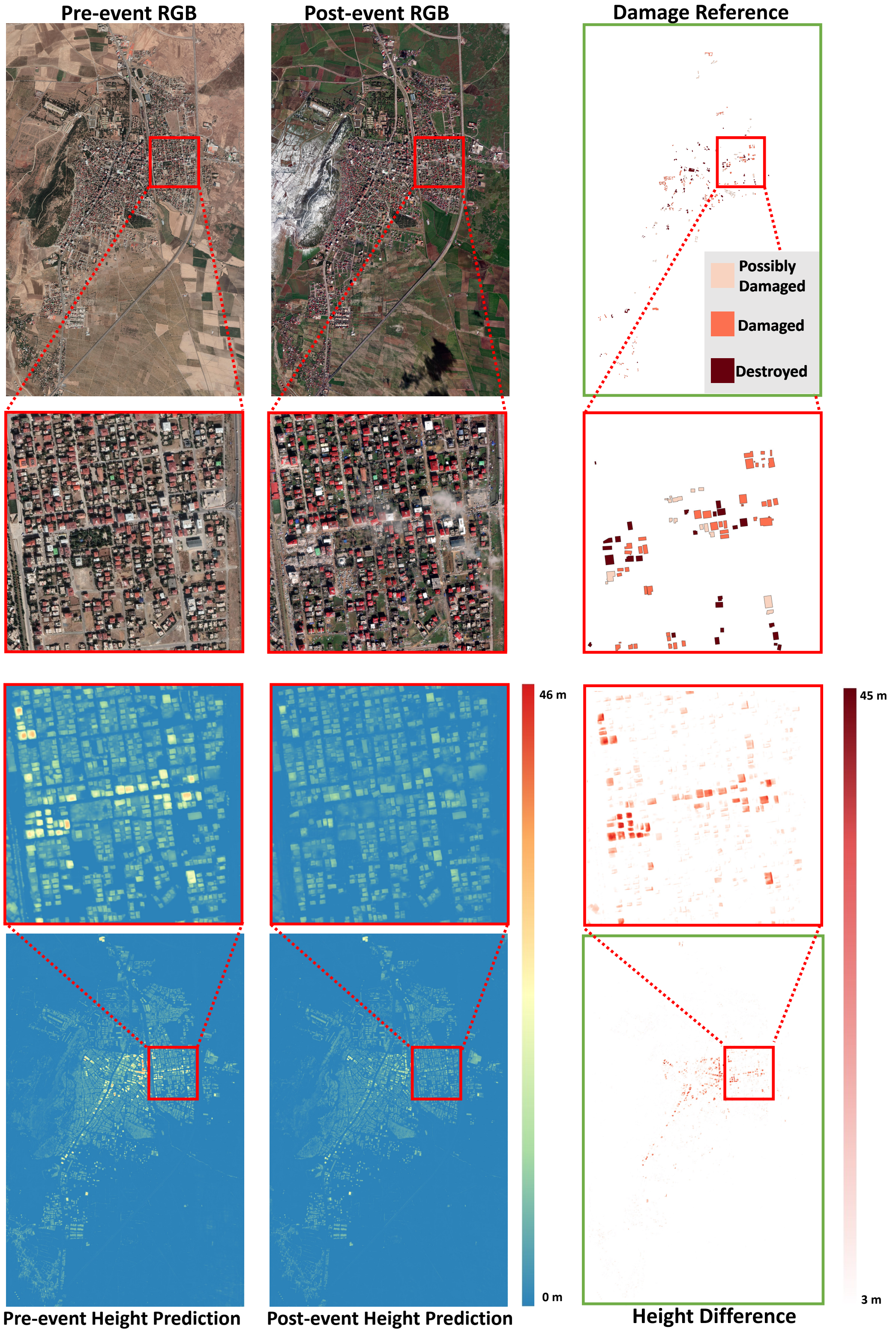


Figure 18: Study case of 2023 Turkey–Syria earthquakes. RGB satellite images: © 2023 CNES/Airbus, Maxar Technologies.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The main claims in the abstract and introduction accurately reflect the contributions and scope of the paper. The abstract clearly outlines the creation of SynRS3D, the largest synthetic remote sensing 3D dataset, along with RS3DAda, a novel multi-task domain adaptation method. These claims are supported by extensive experiments demonstrating the effectiveness of the dataset and method, as detailed in the paper.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The paper includes a Section 6 on limitations where it discusses the appearance gap between synthetic and real data, and the specificity of the RS3DAda method to remote sensing scenarios. This transparency aligns with the guidelines for acknowledging the scope and assumptions of the research.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [N/A]

Justification: The paper does not include theoretical results; it primarily focuses on the dataset creation and empirical evaluations of the proposed methods.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We provides comprehensive details on the dataset generation process in Section 3.2, the RS3DAda method, and the experimental settings, including data splits, hyperparameters, and evaluation metrics can be found in Appendix A. This level of detail ensures that the experiments can be reproduced by other researchers.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: The SynRS3D dataset and RS3DAda code are available at the following links: `https://zenodo.org/records/13905264` and `https://github.com/JTRNEO/SynRS3D`. However, we are unable to provide the code for generating the datasets as some plugins used in our synthesis system are under commercial licenses that do not allow redistribution. Instead, we provide detailed workflows and a list of the plugins used in our system.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: We specify all necessary details about the training and testing setups, including data splits, hyperparameters, the choice of optimizer, and the rationale behind these choices in Appendix A.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: We do not report error bars as it would be too computationally expensive.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).

- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We use one Tesla V100 or A100 for each experiment.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We adhere to the NeurIPS Code of Ethics, taking into account the ethical implications of utilizing synthetic data, and we ensure the responsible application of the data and models created.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: We discuss the societal impacts in Section 6. It highlights the potential benefits for environmental monitoring and disaster response. Since our dataset is synthetic, there are no negative societal impacts, such as privacy or human rights concerns.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [N/A]

Justification: Our dataset and approach do not present these risks.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: For the publicly available existing assets we used, including datasets and code, we have provided proper citations and URLs. Additionally, we will declare certain privately purchased datasets, such as Tokyo and Nagoya, in the Appendix.

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [Yes]

Justification: The new assets, including the SynRS3D dataset and the RS3DAda method, are well documented, with details available in the Appendix and on our GitHub Repository. The documentation alongside the assets, ensuring that other researchers can effectively use and build upon these contributions.

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [N/A]

Justification: We do not involve crowdsourcing or research with human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [N/A]

Justification: We do not involve research with human subjects, so IRB approval is not applicable.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.